TEC-0119

Ascender II – Knowledge-Directed Image Understanding for Site Reconstruction

Allen Hanson
Keith Hoepfner
Edward Riseman
Christopher Jaynes
Howard Schultz
Mauricio Marengoni

University of Massachusetts
Computer Science Department
Lederle Graduate Research Center
Amherst, MA 01003-4610

May 1999

Prepared for:
Defense Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington, VA 22203-1714

Monitored by:
U.S. Army Corps of Engineers
Topographic Engineering Center
7701 Telegraph Road
Alexandria, Virginia 22315-3864

US Army Corps of Engineers
Topographic Engineering Center

T

E

C

U.S. ARMY TOPOGRAPHIC ENGINEERING CENTER
TEC

Destroy this report when no longer needed.
Do not return it to the originator.

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

The citation in this report of trade names of commercially available products does not constitute official endorsement or approval of the use of such products.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	May 1999	Technical 30 April 1997 - 31 May 1998

4. TITLE AND SUBTITLE

Ascender II -- Knowledge-Directed Image Understanding for Site Reconstruction

5. FUNDING NUMBERS

DACA76-97-K-0005

6. AUTHOR(S)

Allen Hanson, Keith Hoepfner, Christopher Jaynes, Mauricio Marengoni,
Edward Riseman, Howard Schultz

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

University of Massachusetts
Computer Science Department
Lederle Graduate Research Center
Amherst, MA 01003-4610

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Advanced Research Projects Agency
3701 N. Fairfax Drive, Arlington, VA 22203-1714

U.S. Army Topographic Engineering Center
7701 Telegraph Road, Alexandria, VA 22315-3864

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

TEC-0119

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This report summarizes research done during the first year of the University of Massachusetts' DARPA APGD contract. Our long-term goal is the development of functional systems for detecting, confirming, and modeling building and other cultural objects from aerial images. The Ascender II system is a knowledge-based system that uses multiple image understanding algorithms and a Bayes network controller that automatically selects and sequences IU algorithms for aerial image analysis tasks. A significant component of Ascender II is a system designed to generate 3-D models of buildings from elevation maps. The system automatically segments a digital terrain elevation map into coherent surfaces and reconstructs a 3-D model of the scene. Finally, Ascender II's range of accessible imagery has been extended to include IFSAR data. A system has been developed that uses knowledge of radar sensor performance at building edges to hypothesize fragments of building boundaries. This information is fused with higher level knowledge about the geometry of buildings, such as constraint on their size, shape, and height, to locate boundary fragments in the image. All systems described in this report have been tested on real imagery, including the Fort Benning MOUT site, Kirtland Air Force Base, Fort Hood, and various ISPRS datasets.

14. SUBJECT TERMS

SAR/IFSAR, Bayes Nets, building reconstruction, image understanding, spatial modelling, site model construction, aerial image analysis, model-based reasoning

15. NUMBER OF PAGES
69

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

TABLE OF CONTENTS

LIST OF FIGURES

Figures **Page**

LIST OF TABLES

Tables

PREFACE

This research is sponsored by the Defense Advanced Research Projects Agency (DARPA) under the Automated Population of Geospatial Databases (APGD) program through Contract DACA76-97-K-0005 and DAAG55-97-1-0188 and monitored by the U.S. Army Topographic Engineering Center (TEC), and sponsored by the Army Research Office (ARO) through Contract DAAH04-96-1-0135 and DAAG55-97-1-0026 (the latter through the Army Research Laboratory (ARL)). The DARPA Program Manager is Mr. George Lukes, and TEC Contracting Officer's Representative was Ms. Lauretta Williams.

ASCENDER II: KNOWLEDGE-DIRECTED IMAGE UNDERSTANDING FOR SITE RECONSTRUCTION

INTRODUCTION

This report presents interim results from the first year of our Automated Population of Geospatial Databases (APGD) research effort on aerial image reconstruction. It is organized into three sections, covering independent yet synergistic aspects of our work. Briefly, the first section covers the structure of the Ascender II system and contains results of recent evaluation and reconstruction efforts on several data sets. The second section details a set of algorithms for recovering (rooftop) surface structure from aerial images. The final section of the report describes our efforts to recover geometric building structures from Synthetic Aperture Radar (SAR) and Interferrometic Synthetic Aperure Radar (IFSAR) data.

One important task in image interpretation is the process of understanding and identifying segments of an image. In this effort, a knowledge-based vision system is being presented, where the selection of IU algorithms and the fusion of information provided by them is combined in an efficient way. Knowledge-based vision systems developed so far have focused on the interpretation problem for a small set of object classes. A major problem with these systems is that the knowledge base, control mechanism, and knowledge sources are combined into a single intertwined system, and the addition of new knowledge or change of domain requires a significant effort. In our current work, the knowledge base and control mechanisms (reasoning subsystem) are independent of the knowledge sources (visual subsystem). This gives the system the flexibility to add or change knowledge sources with only minor changes in the reasoning subsystem. The reasoning subsystem is implemented using a set of Bayesian networks forming a hierarchical structure that allows an incremental classification of a region given enough time. Experiments are presented with an initial implementation of the system focusing on building reconstruction to three different data sets.

Useful representations of the data produced from active and passive range sensing techniques typically require that the 3-D points are segmented into meaningful surfaces and erroneous data are removed. An algorithm is developed in the second part of this report that automatically segments a range image into coherent surfaces and reconstructs a 3-D model of the scene. The technique is composed of a two-phase recursive process. First, a set of points is used to index into a set of surfaces representing the differential geometry of a region. Next, the best set of indexed surface models is used as initial estimates for robust surface optimization in order to converge on the model and parameters that most closely describe the data. After the best-fit surface has been determined for a region, an outlier analysis phase searches for substructures that are recursively processed by the algorithm. The algorithm segments and reconstructs the scene recursively. The technique is demonstrated on two different scenes, both

containing significant amounts of noise, a complex "tabletop" scene of several different objects, and an elevation map of several building rooftops of varying types.

The strength of modern vision algorithms lies not in the ability of any individual algorithm to robustly accomplish its task, but rather in the fusion of information from many sources of data to arrive at an interpretation that represents a consensus of the multiple data sources. The final section of this report deals with the recovery of geometrics structure from SAR and IFSAR data. The approach has two components. Fisrt, one of the edges of the building is found. We locate the building's back edge, which is characterized by the shadows cast in the image. Second, a portion of the building's rooftop is extracted via region growing. This is accomplished by growing outward from the detected edge, iteratively adding rooftop points adjacent to the growing region until no more can be found. Rooftop points are identified using thresholds found in elevation space. We extract enough of the rooftop so that this information, when combined with the attributes of the back edge detected in the first stage, is enough to fully specify the geometry of the building's boundary. The results presented in this report are for buildings with a rectangular boundary, although work is underway for recovering more complex boundary types.

SECTION 1. ASCENDER II: A VISUAL FRAMEWORK FOR 3-D RECONSTRUCTION

1.1 Introduction

The typical knowledge-directed approach to image interpretation seeks to identify objects in unconstrained 2-D images and to determine the 3-D relationships between these objects and the camera by applying large amounts of object- and domain-specific knowledge to the interpretation problem. A survey of this line of research in computer vision can be found in (Haralick and Shapiro 1993).

Typically, a knowledge-based vision system contains a knowledge base, a controller, and knowledge sources. Knowledge representations range from semantic nets in the Visual Information by Semantic Interpretation Of Natural Scenes (VISIONS) system (Hanson and Riseman 1978), and later schemas (Draper et al. 1989), to frames in the ACRONYM system (Brooks 1981), and rules in the System for Photointerpretation of Airports using Maps (SPAM) system (McKeown and McDermott 1985), to relational structures (generalized models) of objects in the MOSAIC system (Herman and Kanade1986). Controllers are typically a hybrid hierarchical system mixing bottom-up and top-down reasoning. As an alternative, heterarchical control systems have been developed using blackboards as a global database. In this case, knowledge sources are triggered if their preconditions have been met, placing their results in the blackboard for future use. Systems developed using this approach include the Address Block Location System (ABLS) (Wang and Shirai 1988) and the VISIONS schema system (Draper et al. 1989).

Recently, vision systems have been developed using Bayesian networks for knowledge representation and as the basis of information integration. The TEA1 system (Rimey and Brown 1992) was developed using a set of Bayesian networks that are combined to define where and what knowledge source should be applied in order to make scene interpretations for a minimum "cost." Although this system used selective perception (Brown et al. 1994) to reduce the number of knowledge sources called, the knowledge base (PART-OF net, Expected Area net, and IS-A net) encoded domain specific knowledge was difficult to construct because of the level of detail required; it had to be re-engineered for a new domain. Although the classification results were satisfactory, it was slow and did not support "real-time" applications (Rimey and Brown 1992). More recently Kumar (Kumar and Desai 1996) introduced a system with simple networks (each network has only two layers) for aerial image interpretation. In this system, after an initial segmentation step, a Bayesian network is built and a set of features, related to each region or to pairs of neighboring regions, are computed from the image. These features are fed into the network and propagated to generate a label for each region. In general, the features are simple to compute but a new network needs to be built for each image.

In most of these systems the controller and the vision algorithms are combined into a single system. One of the problems with this approach is that these systems cannot be easily generalized to different domains from the ones for which they were developed,

and/or the amount of specific knowledge required to use the system in a different domain would be a burden in constructing it.

The Ascender II system was designed for aerial reconstruction of urban areas. It has two major subparts: a reasoning subsystem and a visual subsystem. They are built separately and run on different machines. One advantage of this is that some changes in the reasoning subsystem or in the visual subsystem can be done independently, and available knowledge sources can be exchanged or augmented in the visual subsystem with no changes to the reasoning subsystem.

This section discusses the Ascender II system and presents experiments on three different data sets. The results show that the system is robust across small changes in domain. The reasoning subsystem has a recognition rate of about 90 percent and the set of algorithms used in the visual subsystem produced reconstructions with an absolute error of less than 1.15m in the position of the 3-D vertices. The system can easily be adapted to different environments of the same general type (i.e., aerial images). Section 1.2 presents the system, Section 1.3 shows an example of 3-D reconstruction, Section 1.4 presents results obtained on three different data sets, and Section 1.5 discusses directions for future work.

1.2. The Ascender II System

The original Ascender system (called Ascender I) was developed for building detection and reconstruction of multiple aerial images on a site (Collins et al. 1995). It used 2-D image features and grouping operators to detect rooftop boundaries (Jaynes et al. 1994) and then matched these polygons under 3-D constraints and across views to compute height and build 3-D volumetric models. The system used a fixed strategy and detected nearly 90 percent of the buildings in controlled experiments on imagery from Fort Hood, TX. However, a considerable number of false positives were generated due to scene clutter and the presence of buildings outside of the class for which the system was designed (Collins et al. 1998). In order to address this problem of generality, the Ascender II system has been developed to incorporate AI mechanisms for dynamic control of a large set of Image Understanding (IU) processes; from simple T-junction detectors to complex algorithms, such as the Ascender I system for flat roof building detection.

The Ascender I system demonstrated that the use of multiple strategies and 3-D information fusion can significantly extend the range of complex building types that can be reconstructed from aerial images (Jaynes et al. 1996). The design approach for Ascender II is based on the observation that while many IU techniques function reasonably well under constrained conditions, no single IU method works well under all conditions. Consequently, work on Ascender II is focusing on the use of multiple alternative reconstruction strategies from which the most appropriate strategies are selected by the system based on the current context. In particular, Ascender II uses a wider set of algorithms that fuse 2-D and 3-D information and can make use of Electro-Optical (EO), SAR, IFSAR, and multispectral imagery during the reconstruction process

if available. We believe the system will be capable of more robust reconstruction of 3-D site models than has been demonstrated in the past.

The system is divided into two independent components, running under different operating systems on different machines. The two subsystems communicate through sockets using a communication protocol specifically designed for this application; the system framework is shown in Figure 1-1. Ascender II has been designed as a general-purpose vision system, although our initial effort focused primarily on recognizing and reconstructing buildings from aerial images. Less effort has gone into the knowledge networks and IU processes necessary for other objects, such as open fields, parking lots, and vehicles.

Ascender II assumes that it has as input a set of focus of attention regions in the image data. These regions can be generated in a variety of ways, including human interaction and cues from other sources, such as maps or other classified images. In the experiments described later, our primary goal was detecting and extracting buildings, so the initial regions were generated by using the Ascender I system to detect 2-D building footprints. In one of the experiments described in Section 3, the regions were constructed using Ascender I and a classified SAR image. Once the regions are available, an intelligent control system based on Bayesian networks drives IU processes that extract and fuse 2-D and 3-D features into evidence for a specific set of class labels. Based on accumulated evidence, the system identifies regions as representing an instance of one of the generic object classes defined within the system and, when possible, constructs a coherent 3-D model for each region (Collins et al. 1998).

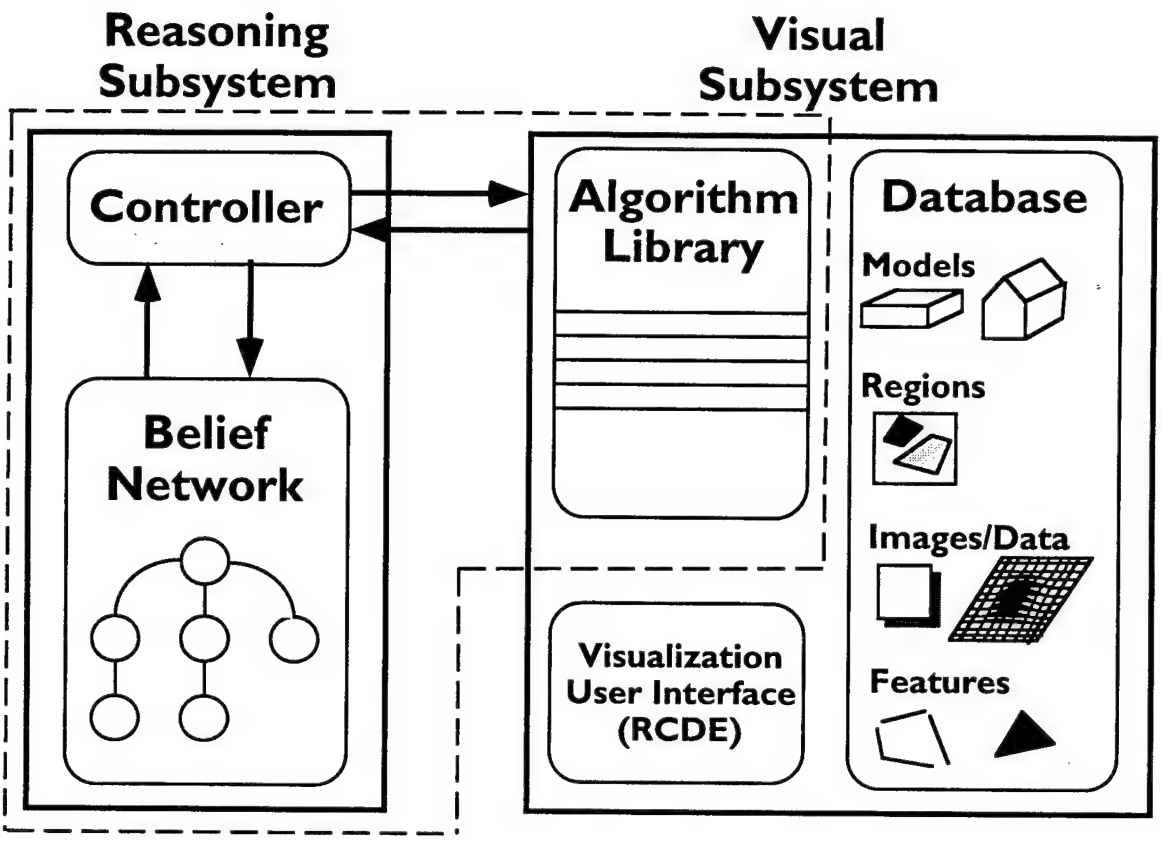


Figure 1-1. Process overview. Control decisions are based on current knowledge about the site. Vision algorithms stored in the visual subsystem gather evidence, update the knowledge base, and produce geometric models.

The reasoning subsystem is implemented as a server and is shown in Figure 1-2. It forks a process for each region and is capable of processing regions in parallel. Each process forked by the reasoning subsystem has the structure presented in Figure 1-1. Note that the regions might not belong to the same image but they should belong to the same image domain. The next two subsections describe each of the subsystems in detail.

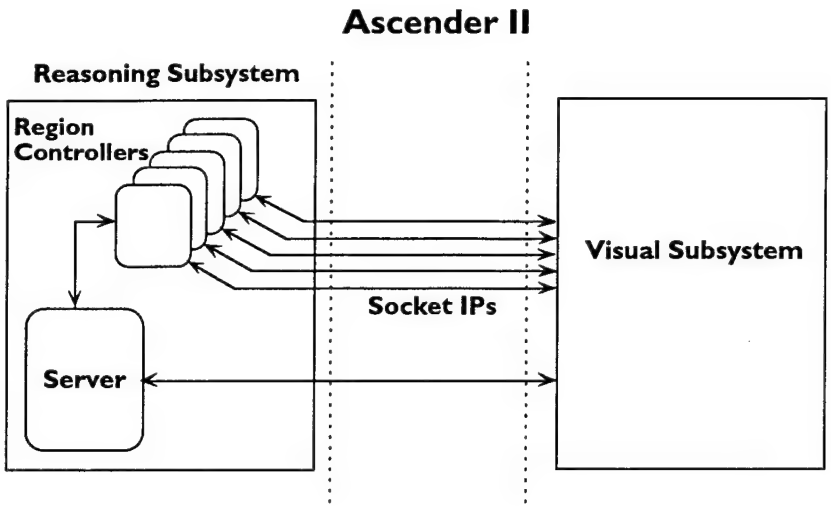


Figure 1-2. The reasoning subsystem starts communication with the visual subsystem and makes a request to find regions in the image. For each returned region the reasoning subsystem forks a process to deal with that region.

1.2.1. The Reasoning Subsystem

The goal of the reasoning subsystem is to accumulate evidence sufficient to determine a plausible identity of the region in terms of the object classes represented within the system. A priori knowledge about objects and their relationships is captured in a hierarchical Bayesian network (see below). Bayesian networks have been successfully used in systems required to combine and propagate evidence for and against a particular hypothesis (Pearl 1988; Jensen 1996). In the Ascender II system, we use these networks to determine what evidence to collect and what to do with the evidence once it has been obtained.

Evidence is obtained by using the network to select appropriate IU processes that obtain relevant feature information from the image(s). A priori knowledge, in the form of initial prior probabilities associated with each object class, is used to select an IU process to use in the initial step. Generally, the process initially selected is fairly generic and measures simple features. In the case of a building, the kind of features measured might include the evidence for a center roof line, the number of L- and T-junctions on the boundary of the region, etc. As evidence accumulates for a particular hypothesis (e.g., a building), the IU process can become much more complex (and presumably return better evidence). Once evidence has been obtained, it is combined with previous knowledge and the process is repeated until the system accumulates enough evidence to determine the region's most representative object class.

The problem with Bayesian networks is that propagation of evidence is, in general, an Nondeterministic Polynomial Time (NP)-hard problem (Cooper 1990) and the time for propagation is a function of the number of nodes, the number of links, the structure of the

network, and the number of states per node (Jensen 1996). It also is known that if the network structure is a tree or a polytree (a structure in which a node can have multiple parents but there is no closed circuit in the undirected graph underneath), the propagation of evidence can be done in linear time (Pearl 1988). In order to avoid the general propagation problem, the reasoning subsystem has been designed using a set of small Bayesian networks organized into a hierarchical structure according to levels of detail. This network structure is the system's knowledge base. A set of decision procedures uses the information inside the networks to decide on what to do next.

Each network represents knowledge about a region at a particular scale of detail. The first level attempts to recognize that a region belongs to a generic class. The second level assumes that the region belongs to the generic class found in the first level and attempts to recognize a subclass that the region can belong to. For instance, if a region is recognized as a parking lot at one level, in the next level it might be recognized as a *full* parking lot using an IU process that counts vehicles in the region and compares this with the area covered. For consecutive levels, the idea is the same, that is, a network at level *i* represents a refinement of a certain class represented in a network at level *i-1*. This organization of networks is shown in Figure 1-3.

The Bayesian networks were developed using the Handling Uncertainty In General Inference Network (HUGIN) system (Andersen, et al. 1989). Our goal is to show that using small networks plus the hierarchical structure as suggested here, increases performance and avoids propagation of evidence through variables that will not affect the overall classification process at a certain level (see Section 3).

The knowledge in each network is structured as follows: each network has only one root node and each state of the root node represents a possible class label for a region. The other nodes in the network are random variables representing features that are expected to discriminate between two or more classes or to help confirm that a region belongs to a certain class. Each arc represents the relationship between features or between a feature and a certain class in the root node. Each leaf node in the network represents a "basic feature." All basic features (and some of the other features represented in the networks) have associated with them a knowledge source (an IU process) in the visual subsystem that is responsible for computing and interpreting the corresponding feature.

Two types of knowledge are encoded in the network: general knowledge about the class (features that are expected to discriminate the class types), and domain-specific knowledge in the form of prior probabilities for each possible class label. Changes in domain imply an adjustment in the set of prior probabilities used for each network in the reasoning subsystem. Because Bayesian networks use Bayes Rule for inferencing, it is possible to reason in both directions. Thus, measuring a feature in the image and propogating its value through the network ultimately changies the belief of a class in the root node for that region.

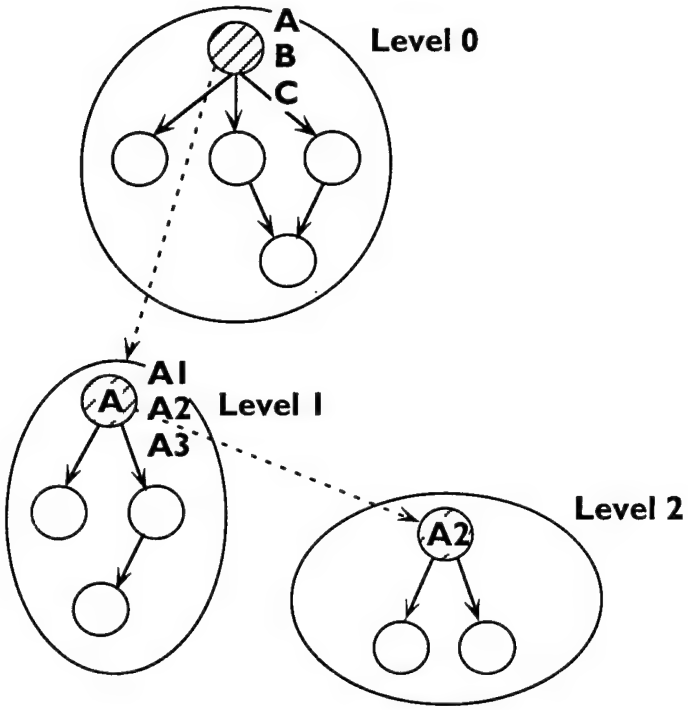


Figure 1-3. The controller starts in level 0 and determines an outcome for the root node at that level (in this case A, B or C). If the outcome is A and time for further computation is available the controller loads the network for A in level 1 (the dotted line shows this inference call). The process can be repeated to subsequent level until the finest level is reached or there is no more time for computation.

The hierarchical structure used in the reasoning subsystem can be used as a basic framework to implement an "anytime" recognition system. Future work will explore this topic.

1.2.1.1. Selecting a feature and the recognition process

A selective perception system performs the minimum effort necessary to solve a specified task (Brown et al. 1994), thus a primary goal of the reasoning subsystem is to select a small subset of features that will allow a consistent identification of the region class. A selection could be made in different ways, e.g., by using a pre-defined static strategy where features are called independent of the evidence acquired, or using a dynamic strategy where a different set of features is called depending on the values obtained for them. Even with a dynamic strategy there are many possible alternatives for selecting a feature. For example, we could simulate the network and select the node with the highest impact on the root node (impact can be seen here as the largest change in the belief distribution of the root node). Alternatively, we could use utility theory as in the TEA1 system (Rimey and Brown 1992).

The reasoning subsystem, as implemented here, uses a simple dynamic strategy and selects the node that has the highest uncertainty. Maximum uncertainty is defined as the uniform distribution over the states in a random variable **N**. A measure for uncertainty, called the uncertainty distance, represents the difference between the value of the maximum belief in the node and the value of the belief if the node has uniform distribution. This measure is computed as:

$$\text{Uncertainty Distance} = \max(\text{Belief}(\mathbf{N})) - 1/S(\mathbf{N})$$

where $S(\mathbf{N})$ represents the number of states in node **N**.

A more classical approach to measure uncertainty is entropy. A simple empirical analysis comparing uncertainty distance and entropy was done. We simulated values for beliefs in a Boolean variable with states **Yes** and **No**. Initially we set the value of **Yes** = 0 and **No** = 1, and then incremented the values of **Yes** under the constraint that **Yes** + **No** = 1. For each step we measured entropy and uncertainty distance. The curves obtained for the **Yes** value using both measures are shown in Figure 1-4; the dotted line is the uncertainty distance and the continuous line is the entropy. One can see that both are symmetric and inversely related; while the entropy decreases (meaning the uncertainty is reduced) the uncertainty distance increases (meaning it is moving from the uniform distribution). In some practical cases, the system using uncertainty distance performed slightly better than the system using entropy. We use this measure in the Ascender II system because we believe that it is more intuitively meaningful.

Given a network, the system computes the uncertainty distance for each node that has a corresponding IU process related to it and selects the node with the minimum uncertainty distance. As an example, consider the case where planar fit and line count both are Boolean variables. If the belief for a good planar fit in a region is 54 percent and the belief for a high number of lines inside the region is 67 percent, the uncertainty is higher for the planar fit variable; therefore this node will be selected.

Once a node is selected, a knowledge source is activated and performs an action on the image. The findings of this action then are returned to the controller, entered as evidence, and propagated through the network. The process is repeated until the controller has enough evidence to recognize the class in which the region belongs. Recognition within the system can be defined in a number of different ways. The simplest is to set a fixed threshold and every time a belief value reaches the threshold, the region is identified as belonging to the class that has that belief value. This measure clearly has a problem: the threshold is fixed but the number of classes (states) in a node is not. If the threshold is too high and the node has many states, it might be difficult to reach the limit. Another approach is to define a relative threshold, for instance when a certain belief doubles from its initial value, or by using a relative comparison between the beliefs inside the node. A third possibility is to use a utility function and keep acquiring evidence until it is not possible to increase the current utility.

For the experiments described later, the decision criteria for selecting a class label for a region in the root node is relative to the other classes at that node. After each new piece of evidence is propagated through the network, the maximum belief and the second highest belief inside the root node are compared. If the maximum belief is at least k times the value of the second highest belief, the controller stops and identifies the region as belonging to the class with the maximum belief. For our experiments, the value of k was set to two. The value was determined arbitrarily, but the results obtained with it were both consistent and accurate; it is unclear how sensitive performance is to changes in k. Ascender II was developed mainly to build 3-D models of buildings and to reconstruct their rooftop geometry. Because of this, finer levels of the network hierarchy were developed for the building outcome of the level 0 network. The networks implemented are presented in Figures 1-5 and 1-6.

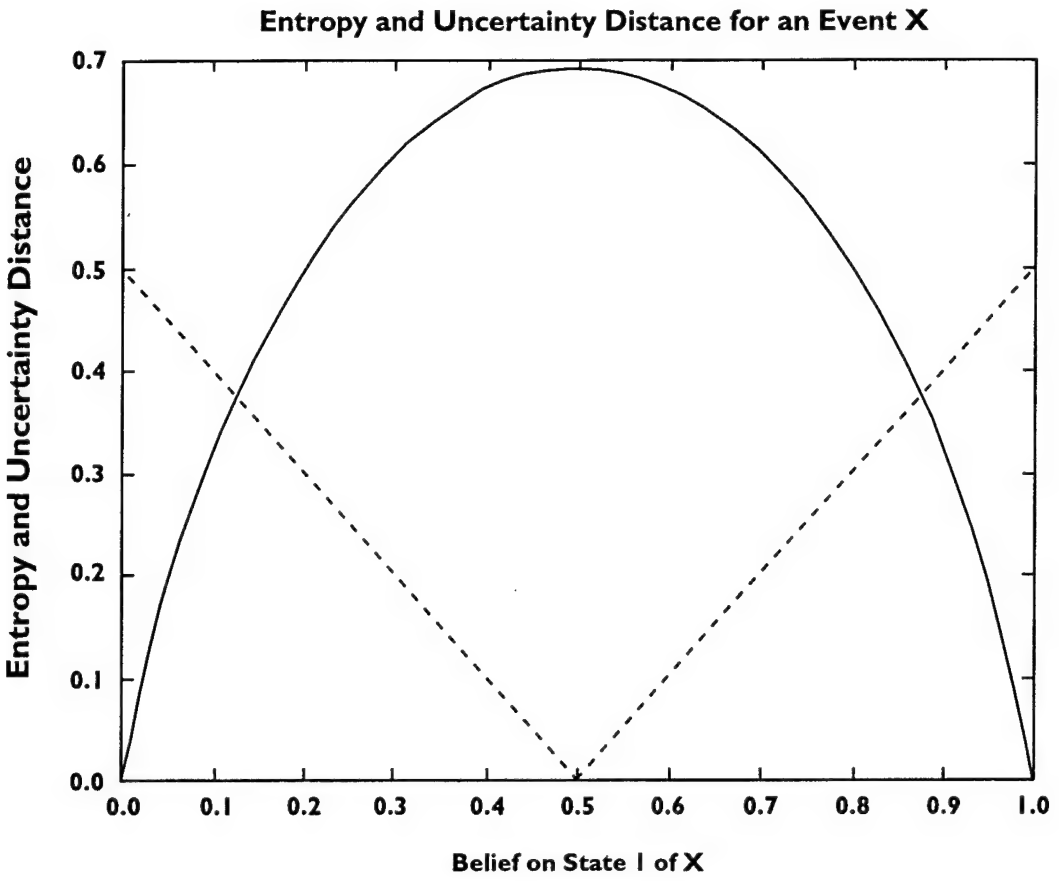


Figure 1-4. The graph above shows entropy (solid line) and uncertainty distance (dotted line) for the state **Yes** in a Boolean variable (see discussion in text).

The network at level 0 (Figure 1-5 left) attempts to recognize the class to which the region belongs. If the class identified is not **Building** the process is stopped, and a generic 3-D model is selected from the database for visualization purposes. In the case where a building is identified, the network at level 1 is called (Figure 1-5 right). This network is designed to identify single-level buildings based on a simple set of features; if a single-level building is identified, the network shown in Figure 1-6 (left) is called to determine the rooftop class. If a multilevel building is identified, there is a possibility the hypothesis is wrong (a multilevel building may have a line in the roof that looks very much like the centerline of a peaked roof). Consequently, the network shown in Figure 1-6 (right) is called to confirm a multilevel building or to backtrack to a single-level building. If a multilevel building is identified, the system breaks the region into two new subregions, based on the evidence gathered, and calls the network at level 1 for each subregion recursively.

Note some knowledge sources can be called at different levels, but because each call is related to a specific region, for which the feature values are stored in the visual subsystem, the system will not compute its value again.

1.2.2. The Visual Subsystem

The visual subsystem is composed of two parts: a function library that stores the set of IU algorithms available to the system, and a geometric database that contains available data in the form of imagery, partial models, and other collateral information about the scene (such as classification of functional areas).

At the request of the controller, an algorithm is selected from the library and run on a region that currently resides within the geometric database. New regions may be produced and stored in the database as a result of processing. In addition, the controller may request that regions be merged, split, or eliminated.

The algorithm library contains information about each of the algorithms available to the system for selection, as well as a definition of the contexts in which each algorithm can be applied. Contextual information, as well as sets of alternative algorithms gathering the same evidence, is stored in the form of an IU process. The IU process encodes the preconditions required for the algorithm to be executed, the expected type of data the algorithm will produce, and the algorithm itself. If the preconditions for a particular algorithm are not met, then an alternative algorithm may be, executed if it is available within the process. If there are no algorithms that can be run in the current context, then the corresponding belief value cannot be extracted by the visual subsystem and must be inferred from the Bayesian network.

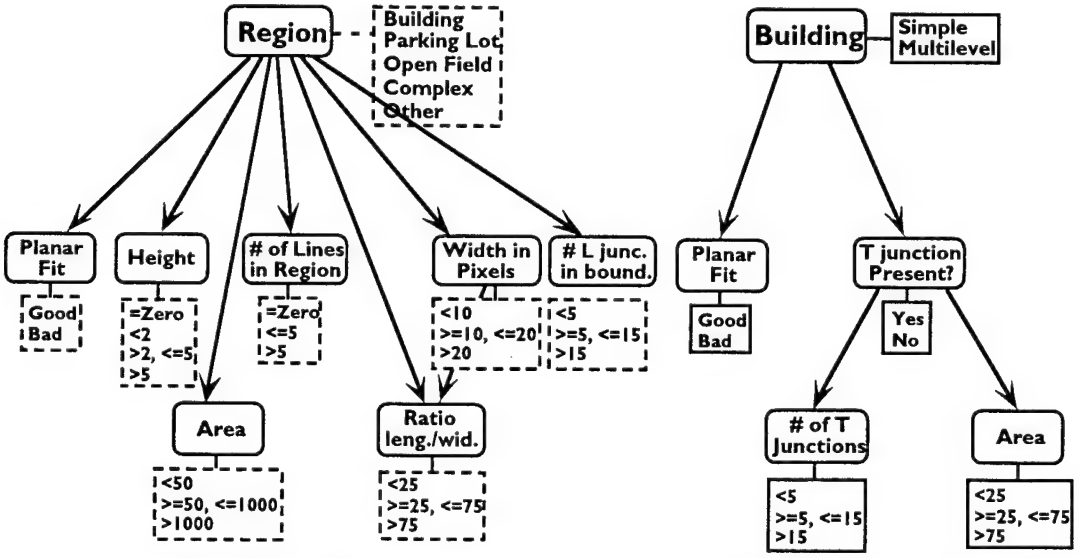


Figure 1-5. The level 0 network at left determines if a region belongs to one of the possible classes. The level 1 network at the right is invoked for each building found in level 0 and tries to determine if it is a simple building or a multilevel building.

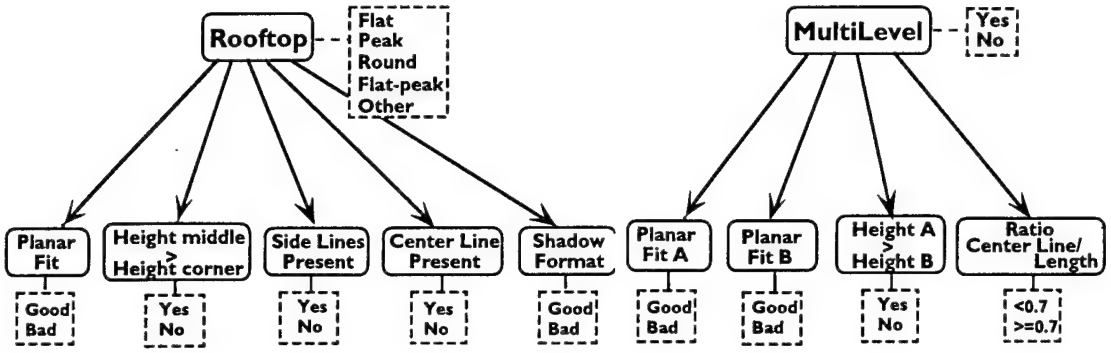


Figure 1-6. Both networks are invoked at level 2. The network at left is called after a single building is detected and is used to determine the rooftop type. The network at right is called when a multilevel building is detected. The reasoning system calls this network to confirm the recognition at level 1.

The library of algorithms was developed to address aspects of the site reconstruction problem from aerial images. For example, finding regions that may contain buildings, classifying building rooftop shapes, and determining the position of other cultural features, are all-important tasks for the model acquisition system. Some of the IU processes may be "lightweight," expected to perform only in a constrained top-down manner, and usable in more than one context. Other algorithms may be complex and contain multiple strategies and associated control; several of the algorithms used to generate the results presented are sophisticated procedures. For a complete list of the Ascender II algorithms see (Jaynes et al. 1998).

If the framework is to be truly general, the cost of engineering a new IU process must not be prohibitive; something that proved to be a problem in earlier knowledge-based vision systems (Draper, et al. 1989; McKeown et al. 1985). Only two components are necessary to convert an IU algorithm into a knowledge source usable by the system: the context in which the algorithm is intended to be run must be defined (Strat 1993), and a method for deriving a certainty value from the output of the algorithm must be defined. This certainty value is used by the system to update the knowledge base using Bayesian inference.

1.3. How the Ascender II system works - Snapshots

This section presents a sequence of snapshots taken when the system was running over a region in the Fort Benning, GA data set. The input image and the regions to be identified are shown in Figure 1-7; this set of regions was obtained using results from processing the optical images by the Ascender I system combined with a SAR classification provided by Vexcel Corp. The prior probabilities associated with the object classes in the level 0 network are shown in Table 1-1.

Consider the sequence of events that occur when the system is processing region A in Figure 1-7. This region is shown in Figure 1-8 (top) with the network at level 0. The

nodes with a thicker ellipse are the ones invoked by the reasoning subsystem. The evidence found and the class identification are shown in the figure. After deciding on a building class, the system called the network at level 1, which checked for T-junctions in the subregions shown in Figure 1-8 (middle). Evidence from level 1 identified the building as multilevel, and the network to confirm multilevel buildings at level 2 is now called. At this point, the system decomposed the region into two new subregions, as shown in Figure 1-8 (bottom), and recursively calls the network at level 1 for each of them.

The process is repeated on each of the two subregions, as shown in Figure 1-9 (top). Region A also was identified as a multilevel building and decomposed into two new subregions. The process again is repeated for the left most subregion A1. This region was not identified as a multilevel building so the rooftop network is called for this particular region in an attempt to identify the type of roof. In this case, the system identifies it as a peaked roof, as shown in Figure 1-9 (middle). The process is now repeated for subregion A2, which also is identified as a peak roof building. The system then considers region B (Figure 1-9 top) and eventually identifies two peaked roof structures, as shown in Figure 1-9 (bottom), which is representative of the final result.

1.4. Experiments and Results

In all experiments described here, only the domain-specific knowledge in the network at level 0 was changed from one experiment to the other (see Table 1-1). This knowledge represents expected frequency for each possible class in the root node and it is represented as prior probabilities. Experiments were performed on the Fort Hood data set (seven views with known camera parameters and corresponding Digital Elevation Maps (DEM)) shown in Figure 1-10 (left), on the Avenches data set (one view and a DEM) shown in Figure 1-10 (right), and on the Fort Benning data set (two views and a DEM), Figure 1-7.

There are six knowledge sources available in the network at level 0 and three at level 1. Table 1-2 presents the average number of knowledge sources invoked by the reasoning subsystem for each data set and also the average calls by region type (object class). The recognition results obtained are summarized in Table 1-3.

The recognition obtained for the regions (Fort Hood data) shown in Figure 1-10 (top) is shown in Figure 1-11 (top). The undecided region (region A) has a belief of 59 percent for the Parking Lot state and 31 percent for the Open Field state (the region is a parking lot). By the decision criteria discussed before, no decision is possible. The misclassified regions are a parking lot identified as a building (region C) and three single vehicles identified as open fields.

The classification obtained for the regions in the Avenches site (Figure 1-10 bottom) is presented in Figure 1-11 (bottom). A set of small regions in the Avenches data set were detected by the Ascender system as buildings and confirmed by the Ascender II system as single buildings with a flat roof; the correct identity of these objects is

unknown. They look like big Recreational Vehicles (RV's) or big containers, as one can see by their shadows and relative size. We considered these areas as RV's that could be identified as either flat roof buildings or a single vehicle. In this case, the system correctly identified them. Region A is a parking lot for boats and was correctly identified as a parking lot. The rooftop of building B was misclassified as a peak instead of flat.

The results for Fort Benning are presented in Figure 1-12. The only complete misclassification was a small peak roof building in front of the church that was classified as a single vehicle. The other problem found was a flat roof building classified as a peak roof building due to a shadow, which generated strong evidence for a center line in that building.

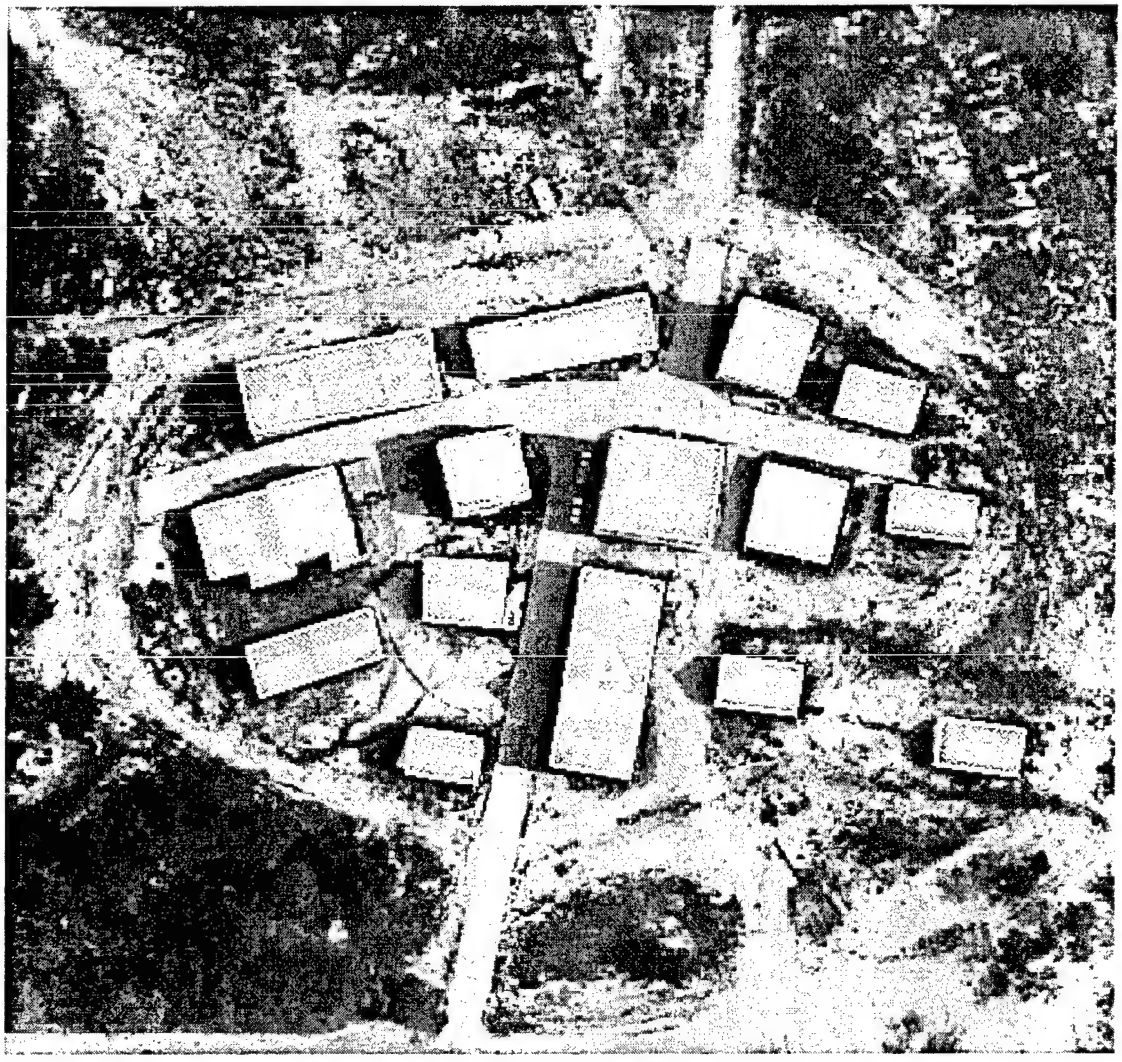


Figure 1-7. Regions to be identified by the Ascender II system from Fort Benning.

Table 1-1. Probability distribution of beliefs in the root node for the level 0 network among the states Building, Parking Lot, Open Field, Single Vehicle, and Other for all three data sets.

	Build.	P.Lot	O.Field	S.Vehicle	Other
FortHood	0.35	0.2	0.2	0.15	0.1
Avenches	0.32	0.25	0.38	0.03	0.02
FortBenning	0.35	0.2	0.2	0.15	0.1

Table 1-2. Average number of calls to knowledge sources for different data sets for all classes (Total column) and by specific classes (remaining columns)

	Total	Build.	P.Lot	O.Field	S.Vehicle
FortHood	3.9	4.25	4.00	3.29	0
Avenches	4.7	5.22	5.00	4.0	0
FortBenning	2.9	3.00	0	0	2.0

Table 1-3. Summary of the recognition process for different datasets. In each case, the number of objects correctly identified is shown, followed by the total number of objects evaluated by the system (* - see text concerning identity of the RV regions).

Dataset	Overall	Level 0	Level 1	Level 2
FortHood	37/41	37/41	21/21	21/21
Avenches	17/18*	18/18	16/16	15/16
FortBenning	17/19	18/19	17/17	16/17

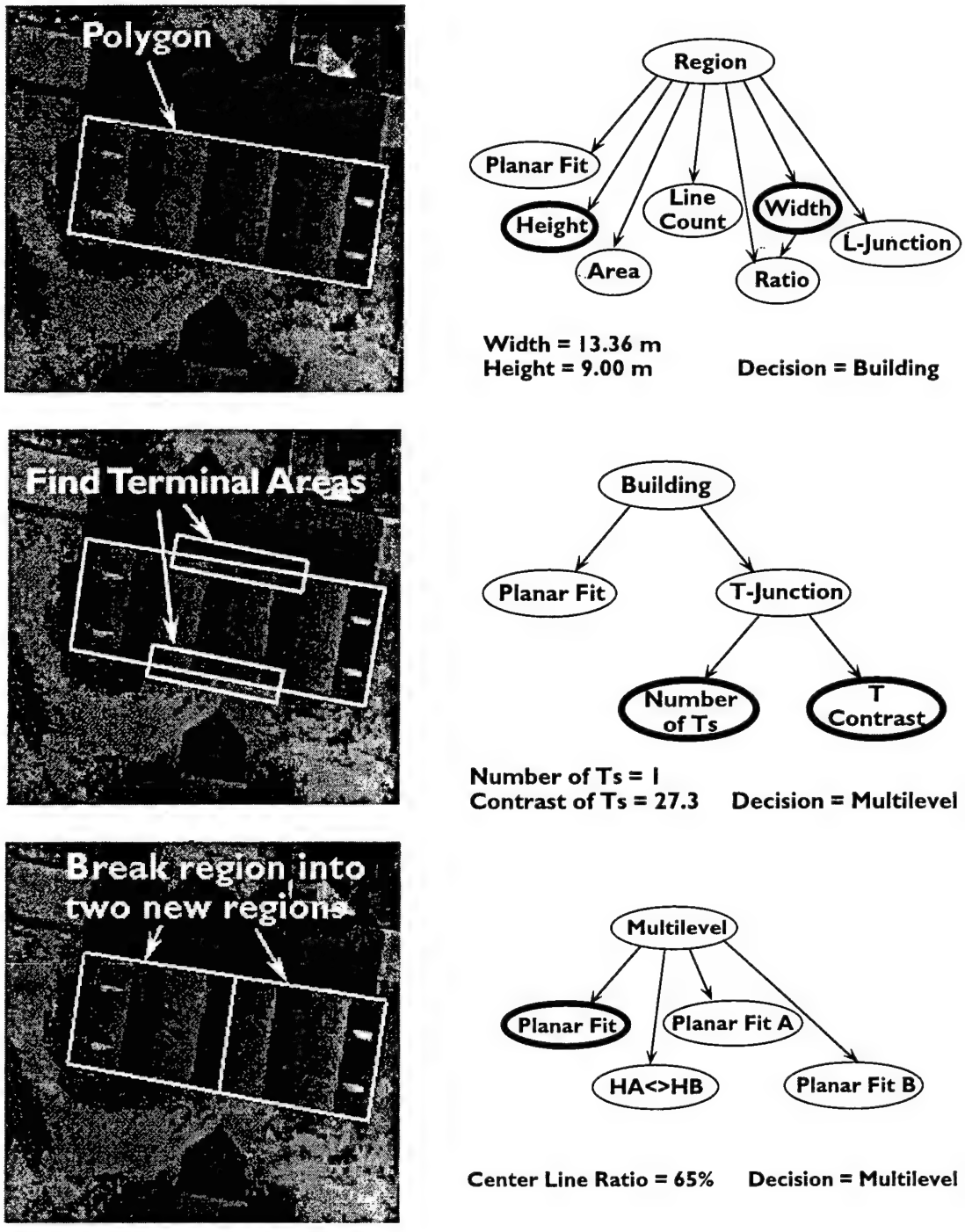
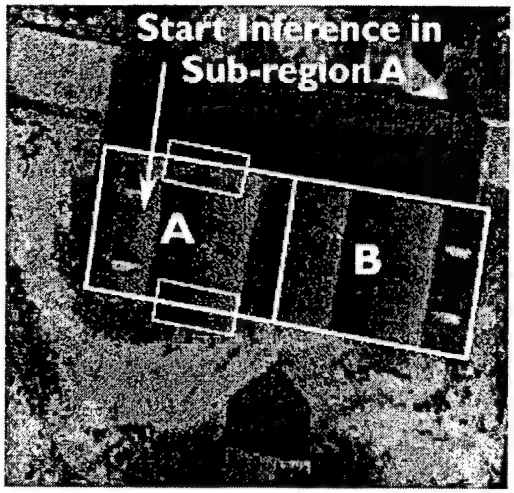
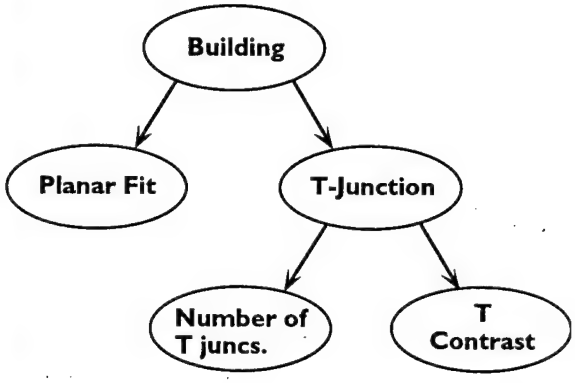
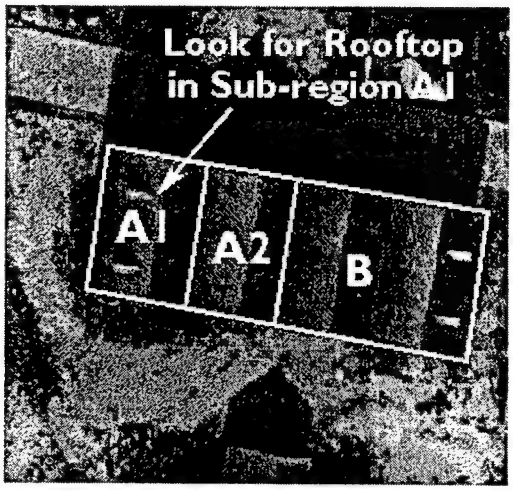
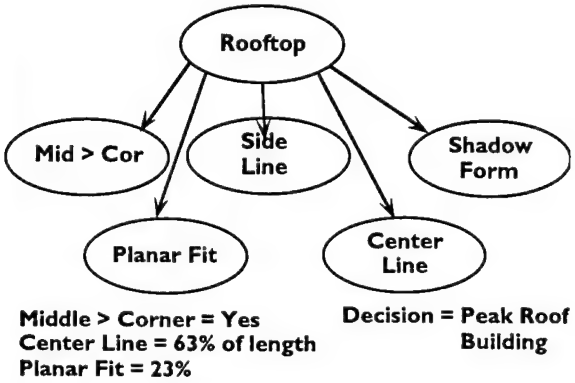


Figure 1-8. Region of discourse. The ground truth is a multilevel building with four peaked roofs. The building class was selected as the most probable because of the combination of the features, Width = 12.36 m and Height = 9.0 m. Middle: The system used the fact that the region was identified as a building, and evidence of T-junctions was found to identify it as a multilevel building. Bottom: The region was confirmed as a multilevel and it was broken into two new subregions.



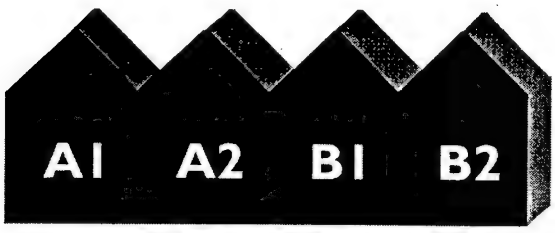






Apply:
1-Peak roof (Correlation = 95.8%)
2-Cylinder (Correlation = 0%)

Reconstruct the region using four peaked-roof buildings



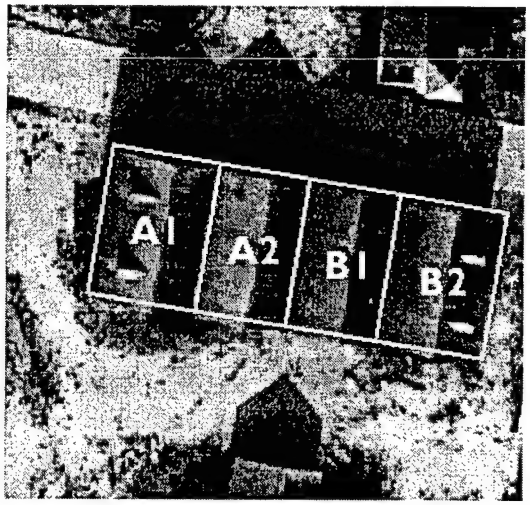


Figure 1-9. Looking for T-junction evidence in the subregions of region A. Middle: Identified region A1 as a peak roof building. Bottom: Reconstructed 3-D model for the whole region.

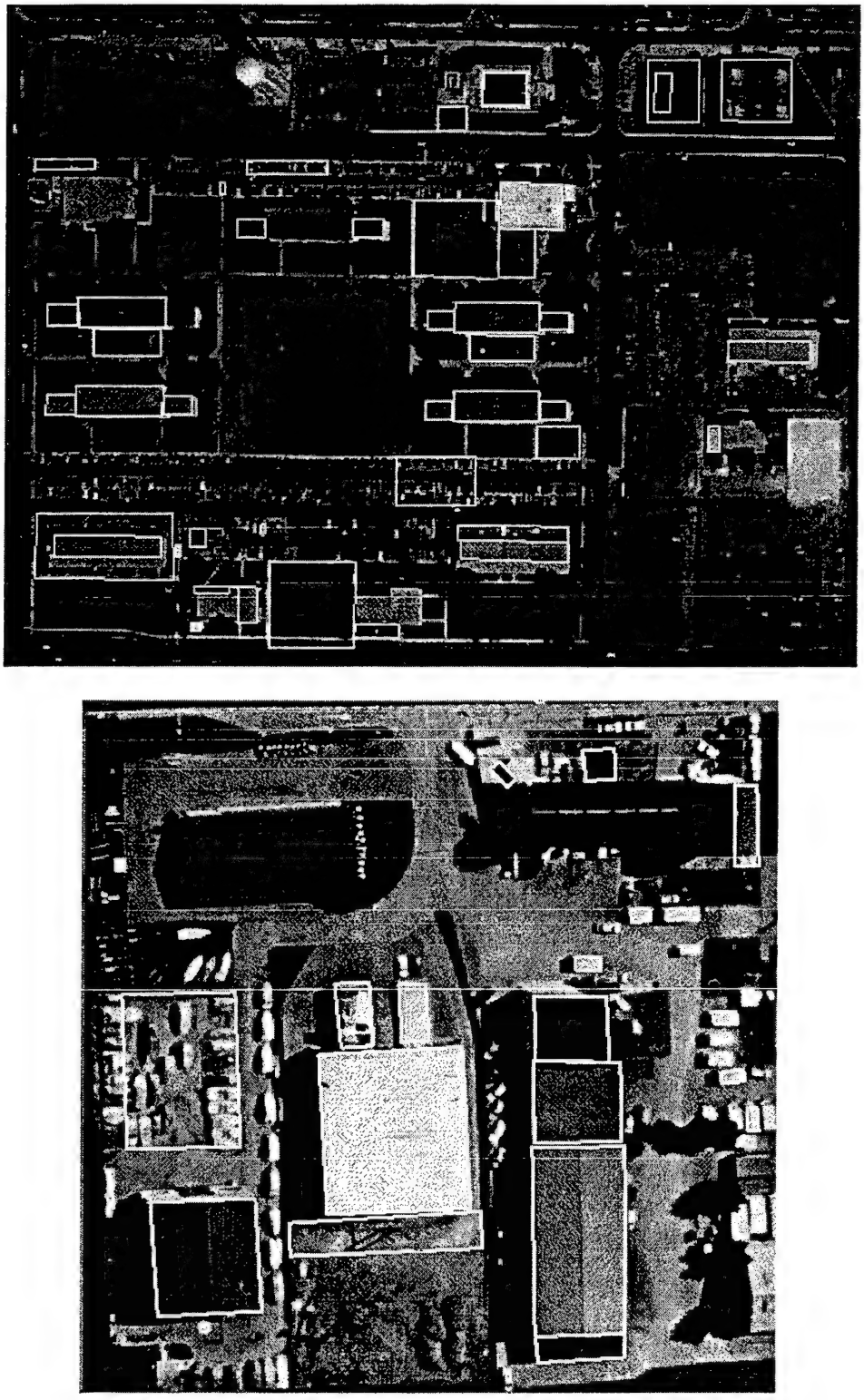
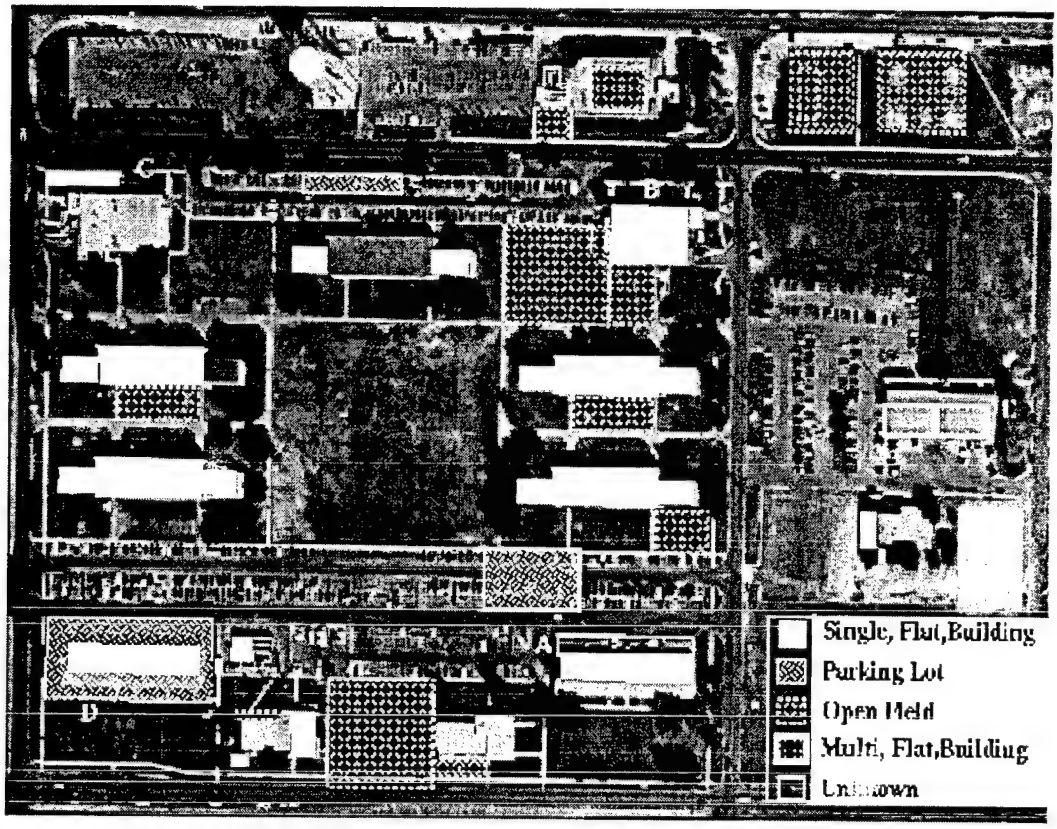


Figure 1-10. At the top: the input regions from the Fort Hood data set. At the bottom: the input regions from Avenches data set. These regions were obtained by running the Ascender I system constrained to detect 2-D building rooftops.



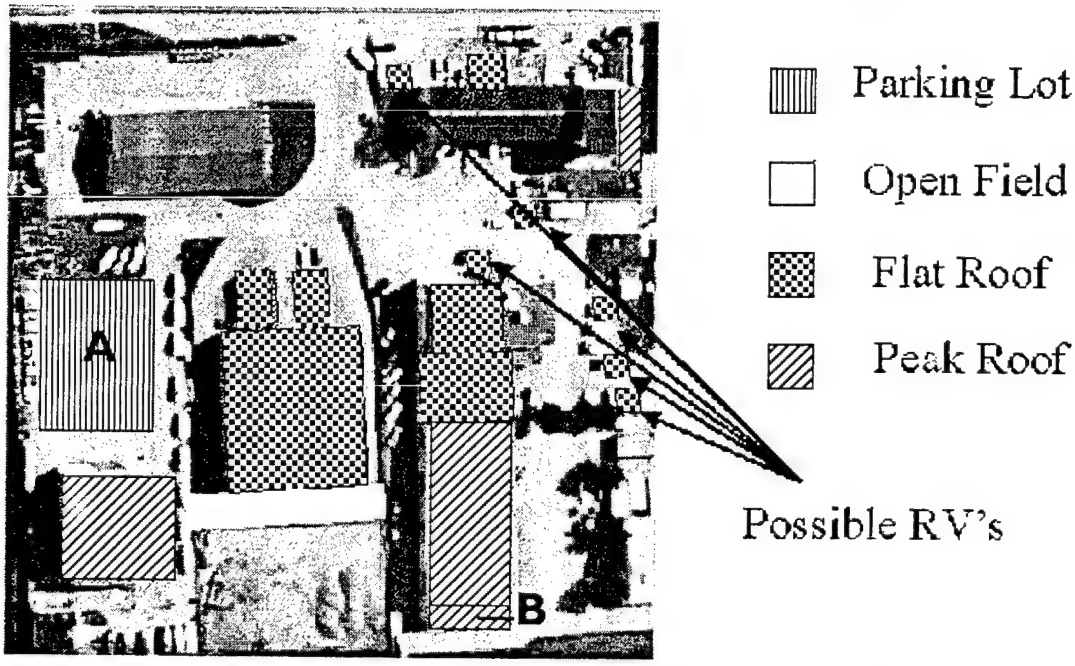


Figure 1-11. At the top: recognition results on the Fort Hood data. Four regions were misclassified and for one region the system was not able to make a decision. At the bottom: recognition results on the Avenches data set.

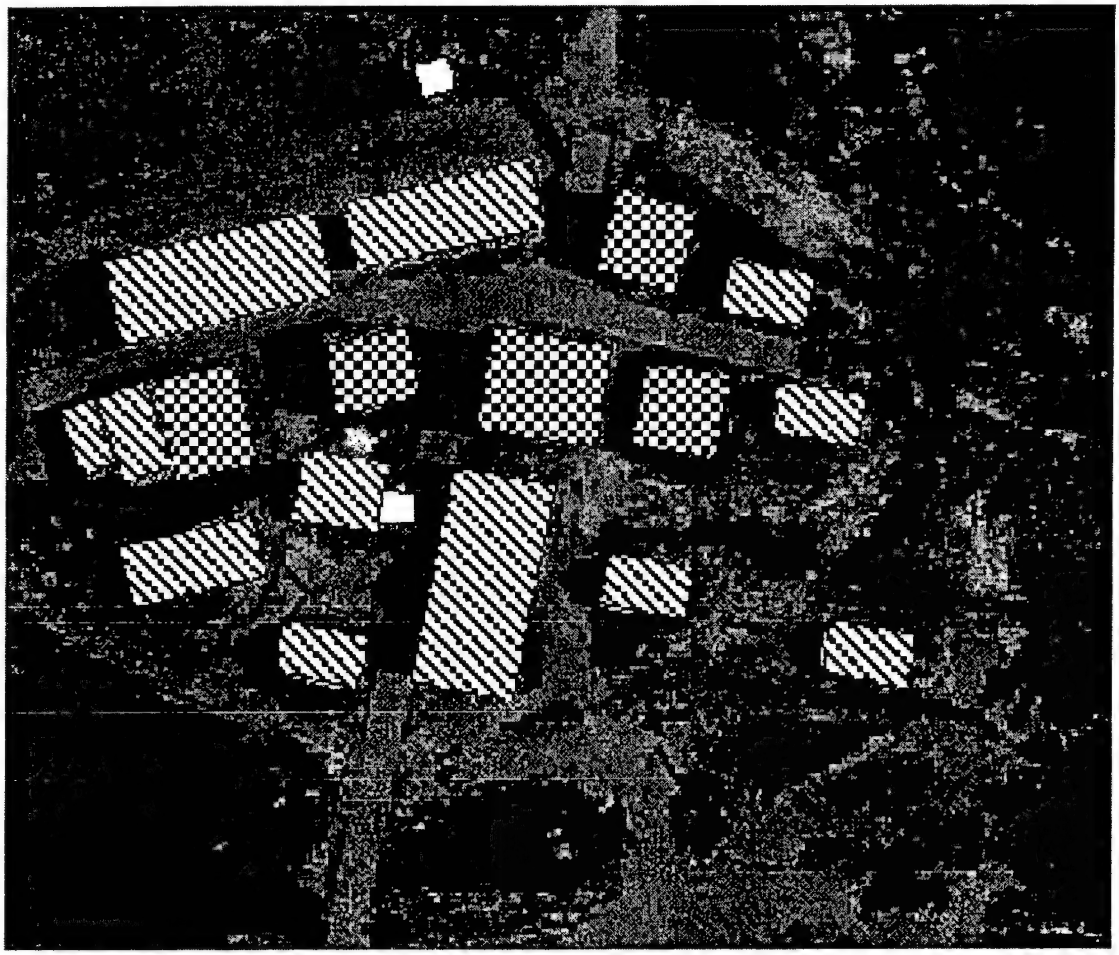


Figure 1-12. Recognition results on the Fort Benning data set.

The final 3-D reconstruction for the Fort Hood data and for the Fort Benning data is presented in Figure 1-13. Geometric reconstruction of building models is based on a robust surface fit to the local DEM using the initial model and parameters generated by the recognition process. Error analysis was performed comparing hand-constructed models for the Fort Benning data, and the errors for the 3-D reconstruction obtained here are shown in Table 1-4 in Planimetric (horizontal), Altimetric (vertical), and Absolute (Euclidean) inter-vertex distances, all in meters.

Table 1-4. Mean, maximum, and minimum errors, in meters, for the 3-D reconstruction of Fort Benning.

	IV Planimetric	IV Altimetric	IV Absolute
Mean	0.528	0.608	0.805
Maximum	0.848	0.892	1.112
Minimum	0.246	0.377	0.524

1.4.1 Evaluation

Our claim is the Ascender II system can perform 3-D reconstruction accurately and efficiently. In order to show that we used the Fort Benning data and tested the Ascender II system against a system that randomly selects a vision operator and also against a system that gets all available evidence prior to making a decision. The summary of these results is presented in Table 1-5. For the system with the random selector, only the best performance is shown. Table 1-6 shows the average number of operators called by each method in the evaluation process. For Ascender II and All Evidence, the time required to process each region on average also is shown in Table #1- 7.

Table 1-5. Number of regions correctly identified and total number of regions at each level for the Fort Benning data set.

Method	Overall	Level 0	Level 1	Level 2
Random	16/19	16/19	14/15	15/15
All Evidence	17/19	19/19	17/18	16/18
Ascender II	17/19	18/19	17/17	16/17

Table 1-6. Average number of operators called for each region for the Fort Benning data set.

Method	Level 0	Level 1	Level 2
Random	3.04	1.31	2.84
All Evidence	7	5	5
Ascender II	2	1.05	2

Table 1-7. Average time of processing on each region for Ascender II and All Evidence methods for the Fort Benning data.

Method	Level 0	Level 1	Level 2
All Evidence	39.6	24.8	1.68
Ascender II	11.3	24.6	0.89

The main difference between the Ascender II system and the All Evidence system in terms of identification was that the small building in front of the church was classified as a single vehicle by the Ascender II system. The system using all available evidence correctly identified it as a building, but misclassified its rooftop as a cylinder and not a peak. The price paid by the All Evidence system was much higher in terms of time required for the correct classification. The random system generated many misclassifications, changing buildings to parking lots and open fields. In the best case, three regions were misclassified in the first level.

1.5. Conclusions and Future Work

A knowledge-based vision system was presented where the reasoning subsystem and the visual subsystem were developed independently. The system is capable of performing

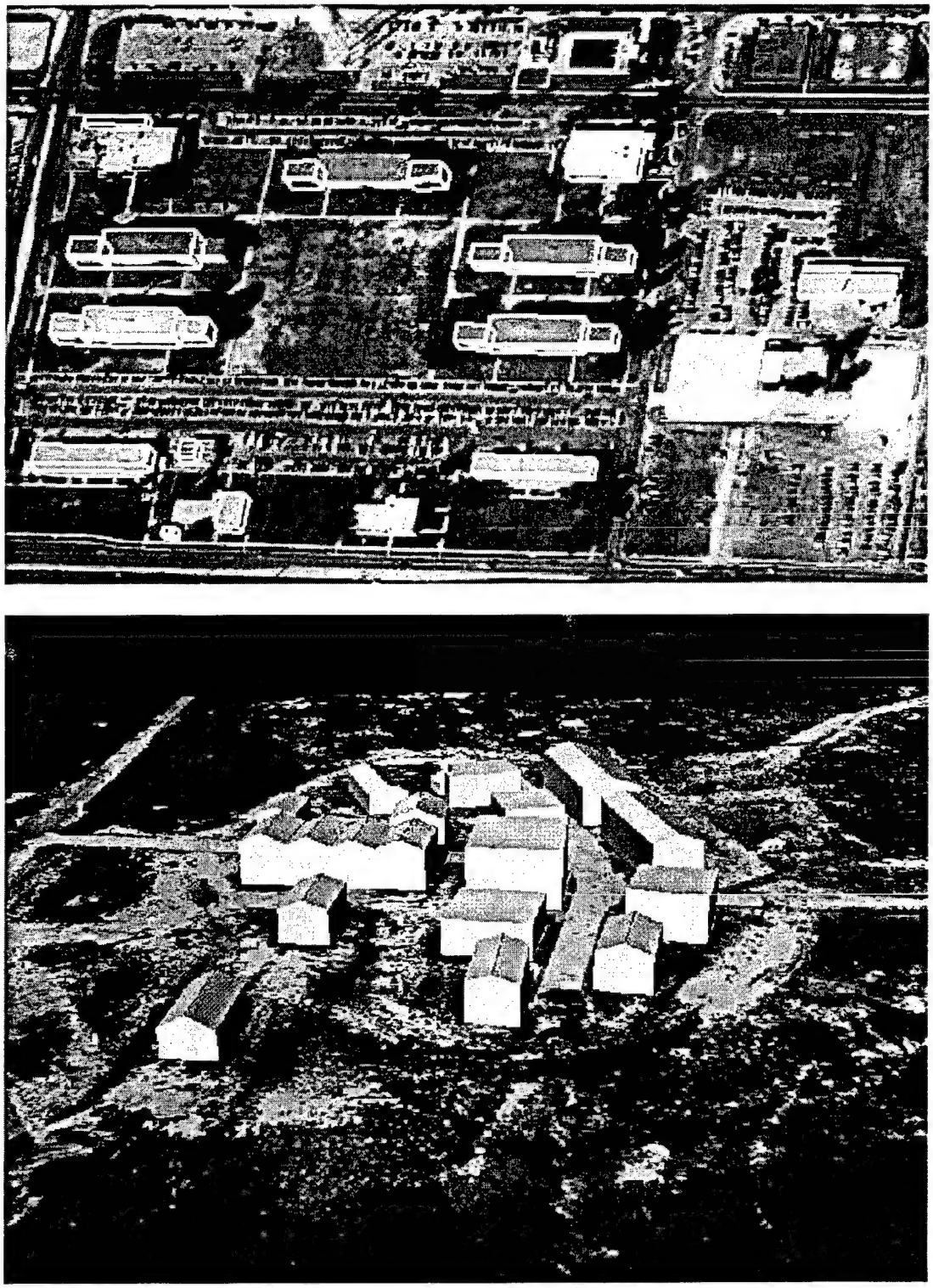


Figure 1-13. At the top: 3-D reconstruction on the Fort Hood data. At the bottom: 3-D reconstruction on the Fort Benning data.

incremental classification of regions. The overall performance is good: 90 percent correct recognition for the Fort Hood data, 94.4 percent for the Avenches data and 89.5 percent for the Fort Benning data. Although the data sets differ in the type of buildings present, image resolution, camera models, number of views available, imaging conditions, etc., a small set of changes in the prior probabilities of the root node in the reasoning system was required to correctly identify the regions in the new site. The hierarchical structure in the reasoning subsystem and the database of IU algorithms in the visual subsystem allow new knowledge and new knowledge sources to be added with minor changes in the system.

The Ascender II system was compared with a system using a fixed strategy and a system using a random selector. In both cases, the Ascender II either performed better or at the same level, but always using fewer resources in terms of either operators or time.

The work presented here is an on-going project. The system is being incrementally developed and new operators are being constantly added. In the future we plan to grow more branches of the hierarchy (i.e., add more object classes) and continue to develop new IU processes. We hope to examine the trade off between hand-coded IU strategies and the dynamic strategies provided by the Bayesian networks. We also will examine alternatives to the simple decision criteria used in the current system. In particular, we plan to develop a utility theory approach to decision making and compare its performance with the current system.

SECTION 2. RECURSIVE RECOVERY OF THREE-DIMENSIONAL SCENES

2.1. Introduction

In this section, we address the problem of both segmenting an unstructured set of range estimates into coherent regions and, for each region, determining the underlying surface. The typical approach to scene reconstruction has been to view segmentation and reconstruction of the scene as two independent problems, or to assume the entire set of range estimates represent a single surface. As a consequence, there HAVE been significant advances in the range segmentation problem (see Besl and Jain, 1985), particularly through surface growing techniques (Besl and Jain 1988, Fua 1995, Taubin 1991, Miller and Stewart 1997) and the problem of model fitting through a number of approaches, including deformable models (Terzopoulos and Metaxas 1990, Kass et al. 1988, Cohen et al. 1991), global model estimation and registration (Zhang, 1994), and mixed approaches (Montagnat and Delingette 1997). Although there has been promising work in surface reconstruction making use of optical images (Jaynes et al. 1997, Fua 1995), and of constraints derived from the formation of a range image from a stereo pair (Fua and Leclerc 1994), the work presented here assumes that corresponding grayscale images are unavailable. See (Jaynes et al. 1997) for similar work under the assumption that both an optical image and an elevation map are available; this paper is attached as an appendix.

The algorithm proceeds recursively in two phases: model estimation followed by model verification. Model estimation indexes into a library of parameterized models using a set of measured 3-D points. The library of models is rank-ordered according to a similarity measure based on the differential geometry of the points. The model verification phase uses the set of parameterized models in the library that most closely matches the measured points as initial estimates for a robust fitting procedure. The model that converges to the lowest residual fit error is used to reconstruct the set of points. Outlier points, with respect to the reconstructed surface, provide a basis for further segmentation, and are clustered into new regions for recursive processing. Regions are removed from the scene in two ways. Either the region is removed during the outlier filtering phase based on morphological constraints, or it is eliminated if a robust fitting fails to provide a sufficient solution.

In order for the algorithm to be successful, two conditions must be met: 1) the scene is composed only of the models in the algorithm's database, and 2) at any one phase of the recursive process, more than half of the points of a region must lie within one of these models. The first requirement is straightforward: the model-directed nature of the problem assumes that models for estimation and reconstruction are available. The second requirement is common to most robust fitting techniques. Both the model-indexing scheme and the final surface fit require that at least half of the range measurements within the region under consideration arise from a single model.

The algorithm may be particularly useful in robotics, for example, to determine both the location and the differential properties of objects for grasping. The recursive nature of the algorithm has applications in high-resolution cartography, where complex building rooftops, containing substructures such as dormers, chimneys, and spires, can be automatically reconstructed.

In order to demonstrate the performance of the algorithm, a range image was generated from a Computer-Aided-Design (CAD) model of a scene containing seven different "Lego" blocks, and two pencils resting on a tabletop surface. Range estimates were produced using a synthetic range sensor, placed directly above the scene, oriented nadir to the table surface. The 3-D points generated from the sensor then were modified using Gaussian noise with μ=0.0and a value of 0.2. This was followed by the introduction of noise in which 10 percent of the (x, y, z) points were randomly perturbed with a Z-value error with a standard deviation of 2.65 cm (1.5 times that of the maximum Z in the scene) and an XY error with a standard deviation of 0.3 cm.

The goal of the algorithm is to simultaneously segment each of the objects from the background and to reconstruct their geometry and corresponding sub-structures. Figure 2-1 shows the rendered model of the scene and a close-up of the 3-D points acquired from a synthetic range sensor placed directly above the scene.

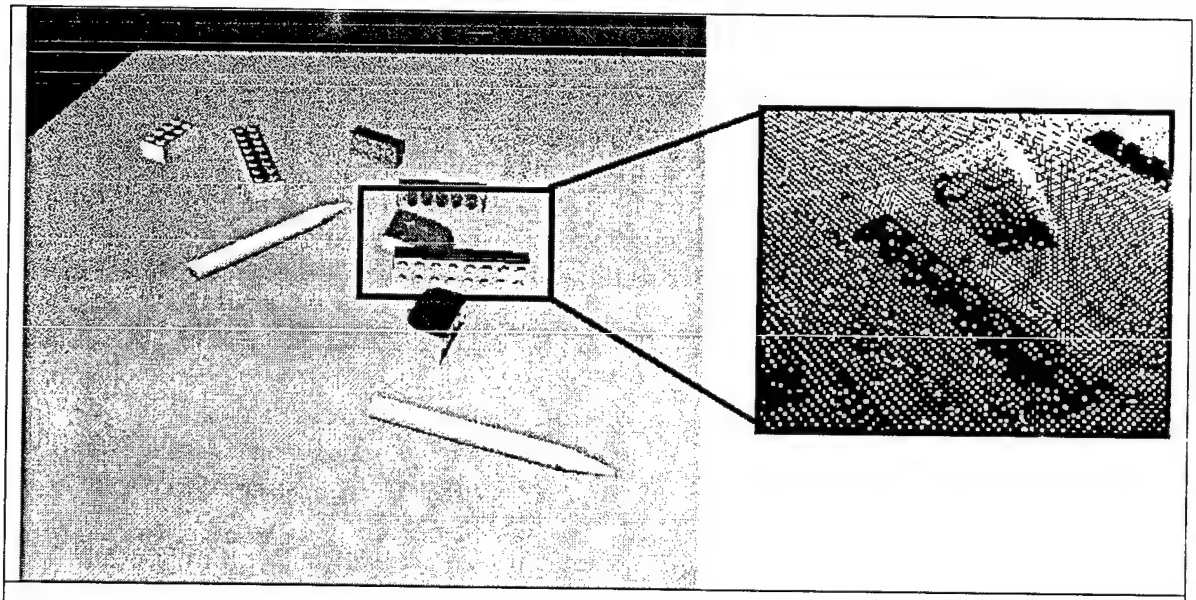


Figure 2-1. Left: Test scene containing nine different objects at various orientations. Right: Close-up of the cloud of 3-D points produced from a synthetic range sensor model corrupted with Gaussian and random noise.

2.2. Model Estimation through Indexing

The estimation scheme indexes into a library of surface primitives based on an analysis of the differential geometry within a region of the range image. The estimated orientations of small patches are used to construct a Gaussian image (do Carmo 1976, Horn 1986,

Zhang 1997) that is correlated with the model library. Correlation provides an orientation vector, and a rotation about that vector at which the histograms correlate maximally. The set of models used for the results in this paper is shown in Figure 2-2. The model library contains seven model classes, and 42 models representing the various possible parameterizations.

Model parameters describe aspects of the model shape itself. For example, the *Peak* model is represented by a distance along a center axis and the angle between the two planes. The number of parameters for each surface in the library is shown at the left of each model in Figure 2-2.

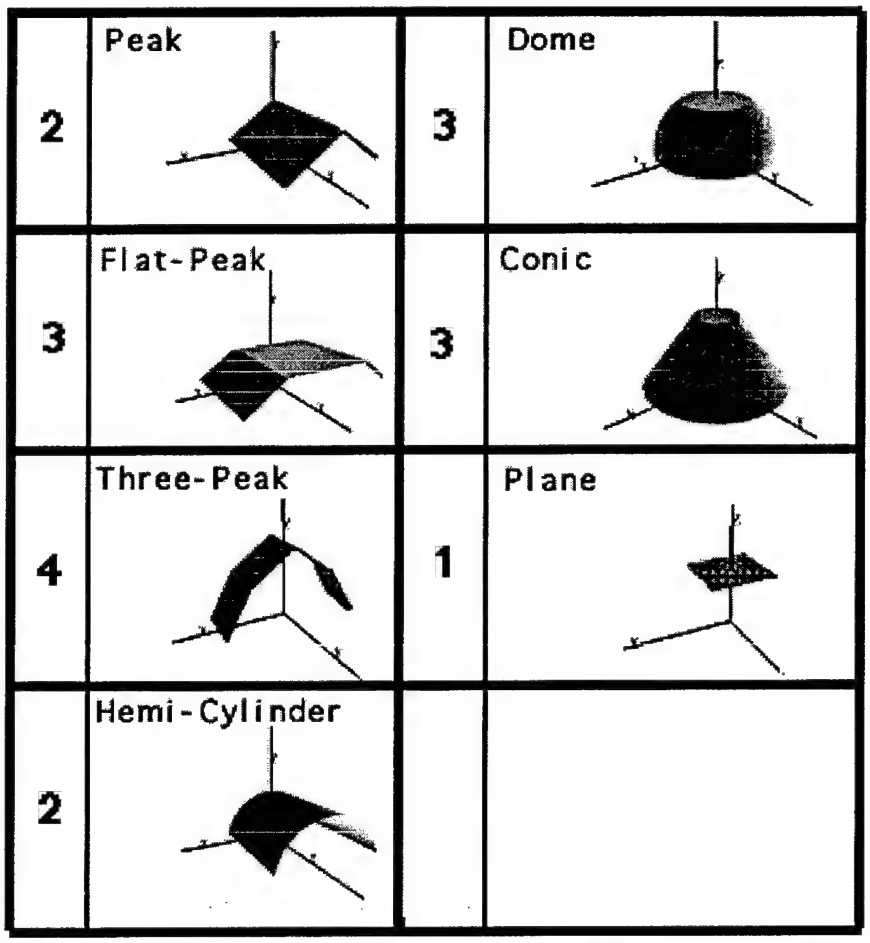


Figure 2-2. Surface model class library. The number of free parameters for each model class is shown at left. For example, the peak model contains two free parameters, its distance along some vector and the angle between the two component planes.

The set of points $x^P = (x, y, z)_k$, within a region *k* of the range image, are triangulated into a simple surface using the Delauney algorithm (Aurenhammer 1991). If no regions are available, as is the case initially, then all points in the range image are used. The computed surface mesh is a set of triangular patches, $T_i = (p_1, p_2, p_3)$, where p_1, p_2 and p_3

are points from the range samples. x_i^P is defined as the point equidistant from p_1, p_2 and p_3 for triangle i. The surface normal for each patch at x_i^P is then computed as:

$$\overline{N_{x_i^P}} = \frac{(p_2 - p_1)}{\| (p_2 - p_1) \|} \times \frac{(p_3 - p_1)}{\| (p_3 - p_1) \|} \quad (1)$$

It is assumed that the vector representing the surface normal pointing "out" from the scene (as opposed to the vector pointing towards the center of objects) is known. This surface normal is used to determine the cell on the Gaussian sphere that will receive a "vote" for a particular orientation.

To avoid sensitivity problems with the method by which the orientation space is subdivided into discrete regions on the sphere, votes are smoothed using a Gaussian distribution. If the surface normal $N_{x_i^P}$, intersects the sphere at $(x,y,z)_s$, the weighted vote is given by:

$$V((x, y, z)_i, B) = c_i \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(D^2/\sigma^2)} \quad (2)$$

where D is the angular distance from $(x,y,z)_s$ to the center of the histogram bucket, B, to receive the weighted vote and c_i represents the area of surface patch i that is contributing to the vote. The amount of smoothing is related to the expected noise in the range image. However, as σ increases, the separability of the model classes degrades. For the results shown here, σ =0.3 and the orientation histogram contains 240 buckets, reflecting a tessellation based on the semi-regular icosahedron (Horn 1986).

A single surface normal may induce a smoothed vote over several buckets, as shown by equation (2), and votes for a given vector no longer contribute when the bucket value of $V((x,y,z)_i, B)$ falls below a threshold (0.1 for the results shown here). Figure 2-3 shows a single region of the "toys" scene (see Figure 2-7b for region labels) after a surface mesh has been fit along with the computed histogram.

To achieve model indexing, the constructed Gaussian image, referred to as the image histogram, then is correlated with each of the model histograms stored in the library. The normalized cross-correlation score is given by:

$$C_{\theta,\overline{O}}(I,M) = \frac{\sum^{(i,j)} (I(i,j) - \mu_I)(M(i,j) - \mu_I)}{(\sigma_I * \sigma_M)} \quad (3)$$

where μ and σ represent the mean and variance, respectively, of each of the image and model histograms.

To select the correct relative orientation of the image histogram and the model histogram, the value of $C_{\theta,\overline{O}}(\mathbf{I},\mathbf{M})$ must be computed for many possible values of _

around several different axes of rotation given by $\overline{O}$. Each of these axes and angles reflect a different alignment between the Gaussian images of **I** and **M**. Prior knowledge about the scene domain (that rooftop models align with the gravity vector, for example) reduces the number of different values of $\overline{O}$ and θ. For the results shown on the tabletop scene, the gravity vector was aligned with the Z-axis and $\overline{O}$ was restricted to 49 different orientation vectors within 30 degrees of the Z-axis above the horizontal plane and θ was restricted to single degree increments about each axis. This allows each model in the database 17,640 different relative orientations between the library model and the extracted histogram.

Each of the models in the library was correlated with the histogram shown in Figure 2-3b. The maximum correlation and orientation parameters for the best three models are shown in Table 2-1. The **Peak,35** model correlated maximally with region 4. Figure 2-4a shows the maximum correlation response for the **Peak,35** model about the 49 different orientation vectors. Figure 2-4b shows the correlation scores for the different values of θ through 360 degrees about $\overline{O}$; the maximum correlation was found at θ=2.09.

Table 2-1. Top three models matched to the region shown in Figure 2-3a. No value for θ is reported for the plane model because it is circularly symmetric. All three models are fit to the region to determine the appropriate reconstruction.

Model Name	Correlation	$\overline{O}$	Rotation Angle θ
Peak,35	0.836	(0.38, 0.0, 0.92)	2.09
Peak,25	0.797	(0.13, 0.0, 0.99)	2.13
Plane	0.664	(0.18, 0.18, 0.97)	

2.3. Model Verification

Model indexing provides an ordering over the set of models $M_i(x;\bar{a})$ and associated parameters within the model library for a set of points within a region of the range data x^P. The parameter vector $\bar{a}$ and the model M are used as initial estimates for a robust surface fitting procedure. The top several models are fit to the data points and the model that converges to the best fit is used to interpret the data.

Surface fitting involves a multidimensional optimization scheme for $M(x^P,\bar{a}) = 0$ where $\bar{a}$ is the parameter vector associated with the model being fit. Because a triangular mesh has already been fit to the range data, the surface normal at each patch $\overline{N}_{x_i^P}$ is used to compute the distance between the current model and the observed data. Specifically, the median of

$$E_P[\left|\hat{x}^P - x^P\right|^2] \qquad (4)$$

is minimized, where

$$\overline{x^P\hat{x}^P} \mid \hat{x}^P = t \bullet \overline{N}_{x^P} + x^P, \quad \hat{x}^P \in M(\bar{x};\bar{a})$$

that is, $\hat{x}^P$, is the point on the fit surface corresponding to x^P and is obtained along the computed surface normal. This median squared error function avoids measuring error in an arbitrary way, and uses information from the surface mesh to estimate an appropriate direction from the observed data to the model surface. This is particularly important for models with sharp surface discontinuities (the peak model, for example), where error measured near the peak and along the Z-axis may induce an unusually large error.

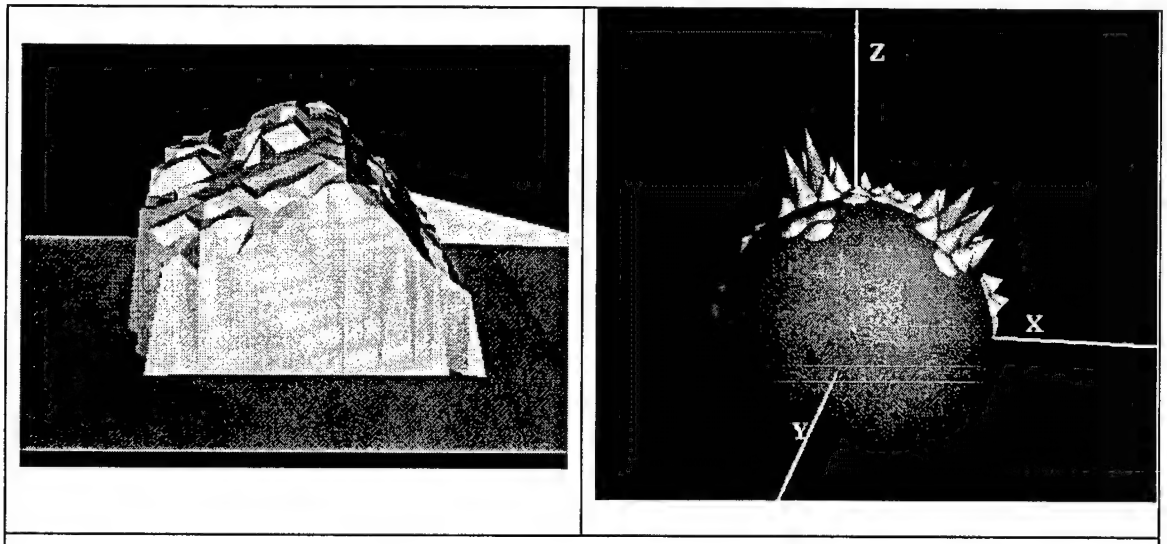
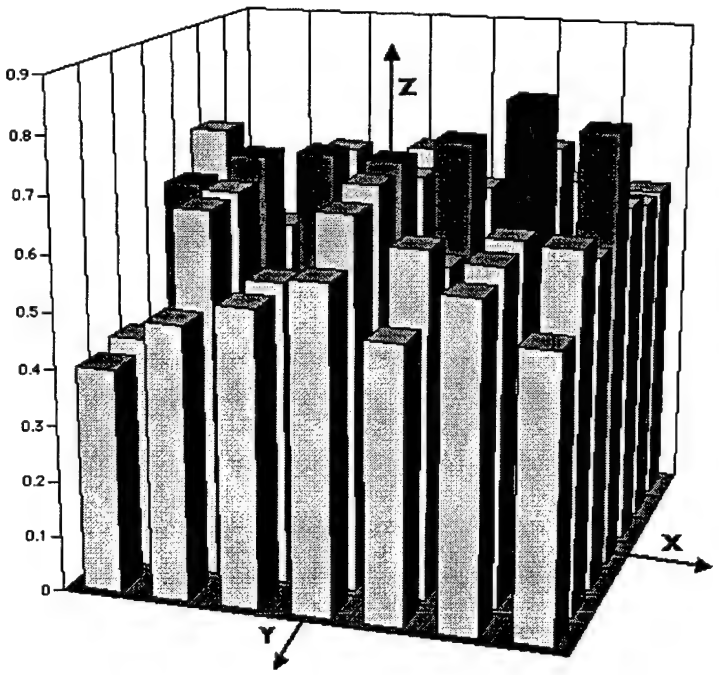


Figure 2-3. (a) Surface mesh fit to region 4 of the toy scene. The object is a tilted 4-peg Lego block. (b) Constructed Gaussian image.

A multidimensional simplex method (Nelder and Mead 1965) is used to minimize equation (4) over the ***k***-dimensional space induced by the number of free parameters in the selected model. To avoid optimization over a large number of parameters, neither position nor absolute rotation is part of $\bar{a}$. Note that absolute rotation is computed as part of model indexing from the computation of θ and $\overline{O}$ as the vector at which there is a maximum correlation response between the two histograms. Absolute position in the scene is fixed as the center of the region of data being fit; therefore, models are restricted to move along $\overline{O}$. For example, the plane model has one free parameter after model indexing – its distance along $\overline{O}$.

Outliers are computed as points in the range data that have a relatively high residual error. Because the outlier measure, with respect to the model $M_i(x;\bar{a})$, is the basis for the segmentation of new surfaces, it is important that outliers are not computed from a simple error prone threshold on E_P. Instead, outliers are computed on the fly, through multiple fits using the simplex method. At each iteration, the points with the largest error measure are discarded as outliers, leaving ***k*** inlier points for a new fit using the same procedure.



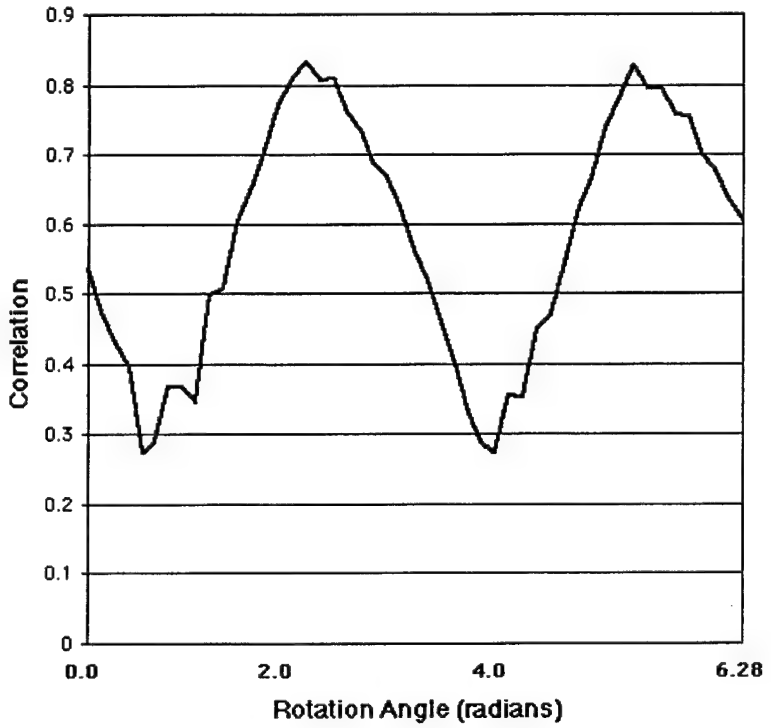


Figure 2-4. Maximum response of correlation score about the 49 different axes of rotation. The rotation vector (0.383,0.0,0.924), shown as a darkened column (top), correlated maximally with the data with a response of 0.82 (shown in the bottom graph).

A χ^2 per degrees of freedom measure[1] is used to determine when discarding outlier points no longer improves the surface fit:

$$\frac{\chi^2}{k-1} \tag{5}$$

where

$\chi = \sigma(E_p)$
k = number of inlier points

When the value of equation (5) does not decrease as more outlier points are removed from the data, the process stops. Using this technique, the number of outliers removed at each step can be small and is not dependent on characteristics of the data, as a simple threshold based on E_p would be. Figure 2-5 shows a close-up view of object #4 in the tabletop scene and the reconstructed surface obtained by fitting the **Peak,35** model to the data by minimizing the least median error as described above.

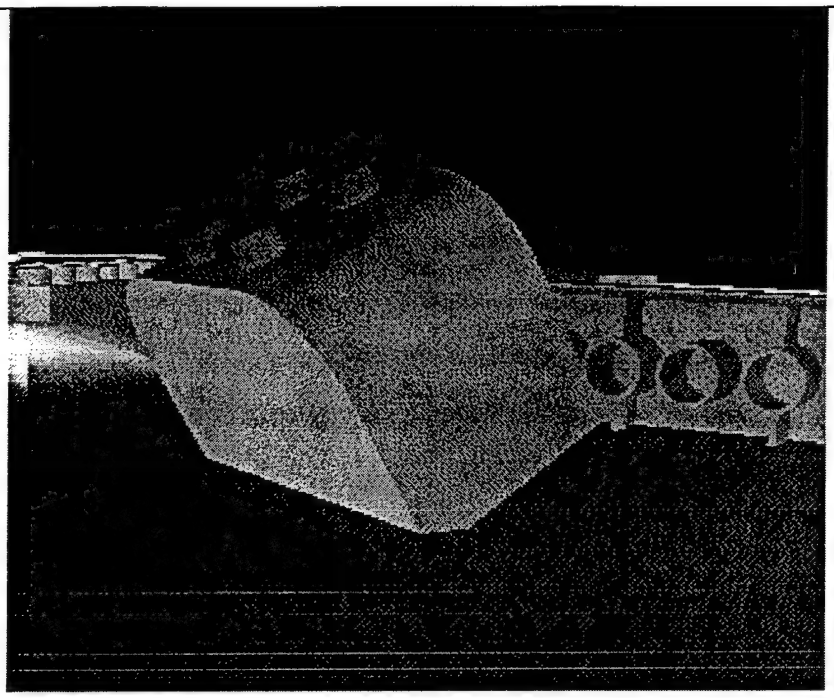
2.4. Outlier Clustering

After a surface has been fit to the data using the procedure described in the previous section, data points are classified as either inliers or outliers. Outlier points are then clustered into spatially coherent regions and the algorithm is recursively applied to the extracted regions.

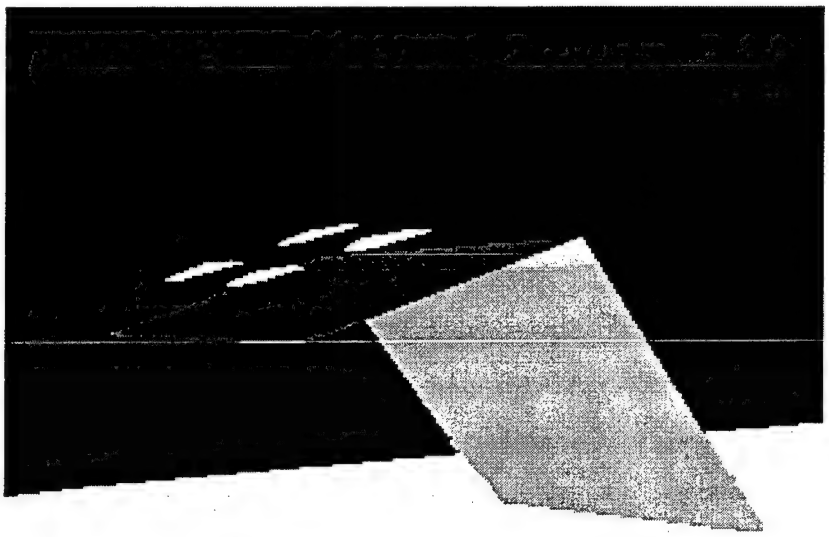
Production of valid outlier regions is a straightforward, three-step morphological process. First, a closing operator creates connected component clusters in the range image. An opening operator then removes small sets of residual points due to noise. Finally, a dilation step creates complete connected regions. Each region is discarded based on a size constraint that can be derived from the expected minimal size of objects and the known sensor model.

Figure 2-6 shows the outlier points with respect to the peak model fit to region 4. Outliers are due to noise in the range data, inaccurate model fits, and substructures present in the scene. Although object 4 is curved near the side boundaries of the top face (see Figure 2-5a) the library contains no such surface and the peak model was fit. This produces the long bands of outliers (Figure 2-6a) near the boundaries. Figure 2-6b shows the remaining regions after outlier clustering that are recursively processed by the algorithm.

[1] Originally suggested by Howard Schultz via personal communication.}

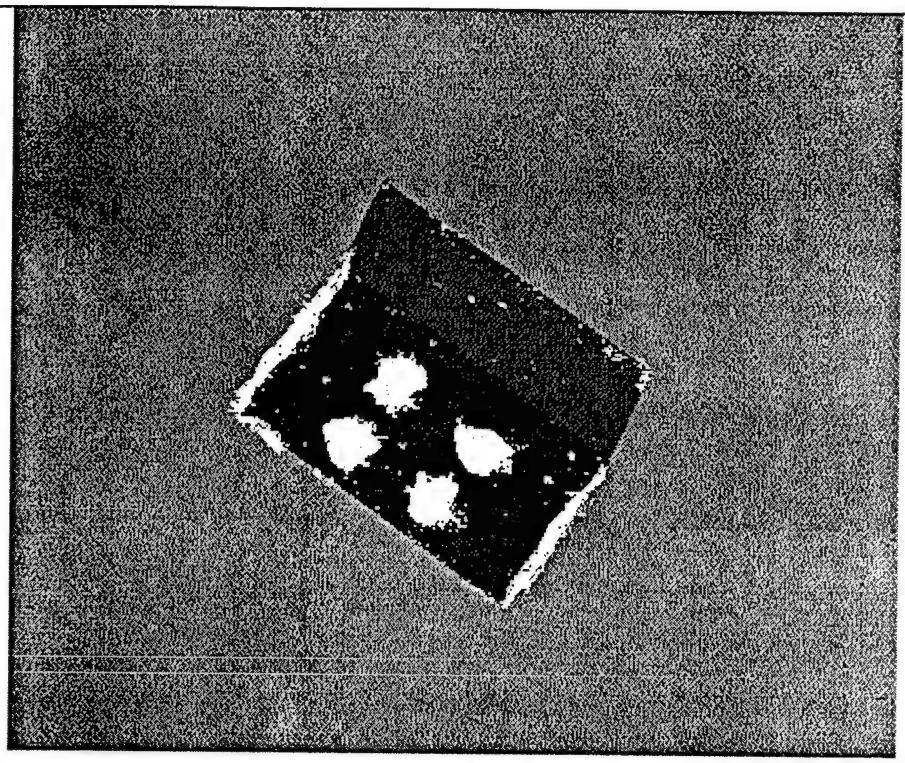


(a)

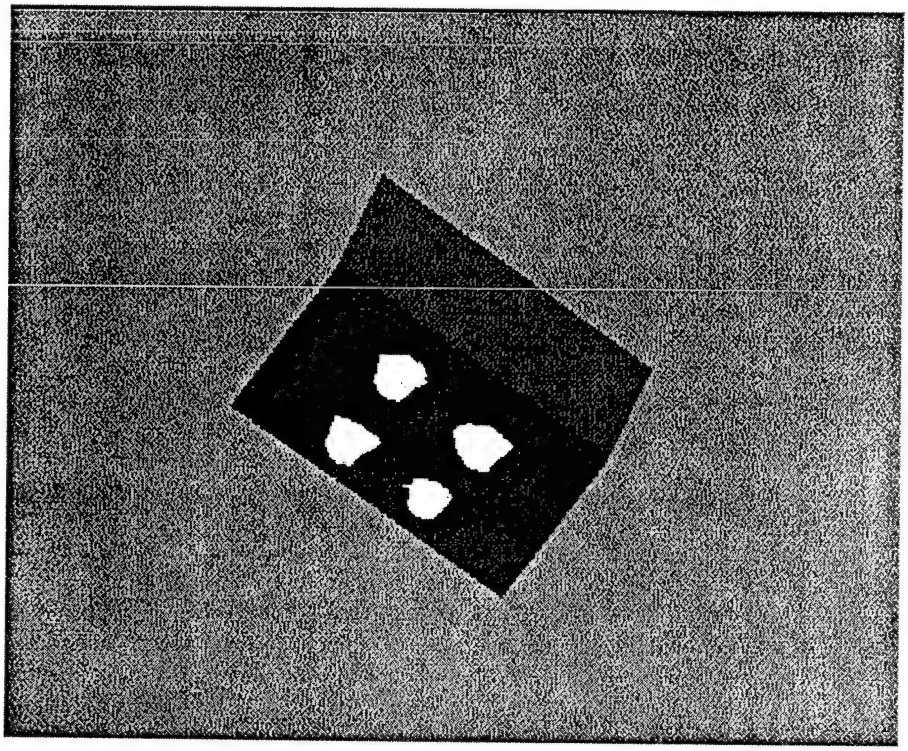


(b)

Figure 2-5. (a) Close-up view of object #4. Note: The object occluding object #4 has been removed to allow a clear view for comparison. (b) Reconstructed surface of region 4. Note that subregions have been detected and reconstructed (see outlier clustering).



(a)



(b)

Figure 2-6. (a) Outliers with respect to the model fit within region 4. (b) Remaining outlier regions after clustering.

2.5. Results and Conclusions

The algorithm was run on two different scenes. Because the "tabletop" scene was generated from ground truth models, it was used to study the accuracy of the algorithm. Another test was run on the Ascona ISPRS, "flat" scene; specifically, the elevation map of several buildings that was produced from a stereo optical routine was used as the input data.

2.5.1 Tabletop Experiment

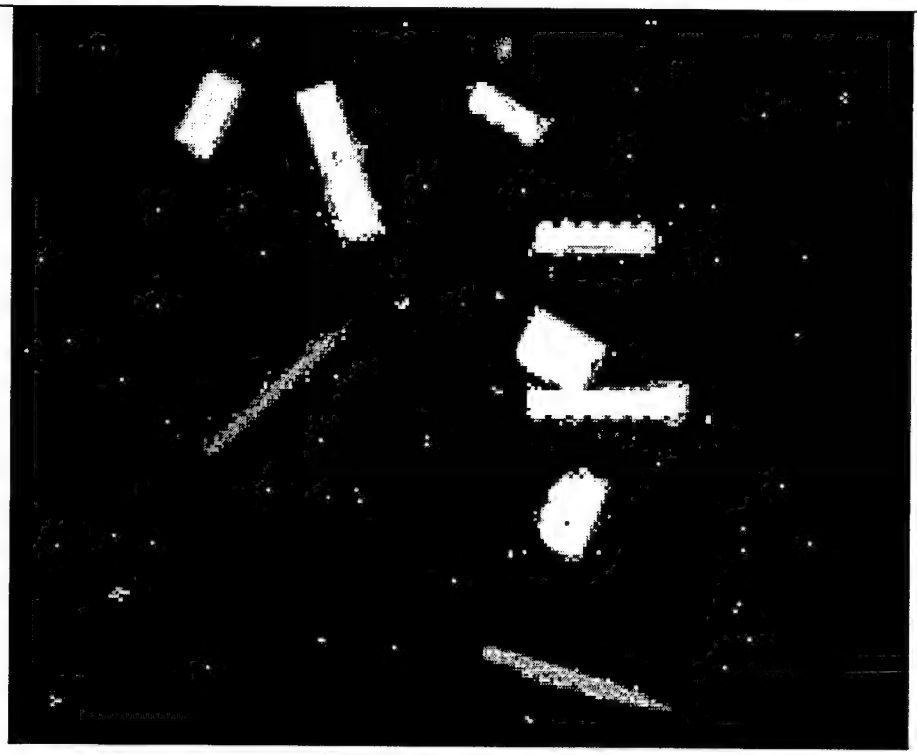
Figure 2-7a shows the actual range data used to reconstruct the tabletop scene. The image is 512x512 pixels with a spatial resolution of 12.36 samples per centimeter. The synthetic range sensor was perpendicular to and located above the table surface. Initially, the algorithm recognized a plane and reconstructed the table surface. All outlier regions, with respect to this fit, were then discovered and clustered. Each of the remaining regions, after the algorithm terminated, are labeled and shown in Figure 2-7b. For each of the regions shown, new outlier regions may have been produced and reconstructed. These were all correctly detected as planar segments above the objects. Two subregions within region 2 were reconstructed as a single surface. As the number of points within a region becomes small, the clustering algorithm is sensitive to the presence of noise and can merge regions located near one another.

Figure 2-8 shows the reconstructed scene. The scene is a set of recovered surfaces in the world coordinate system. Of course, the hidden surfaces (with respect to the range sensor) are unknown and are not part of the reconstructed scene. Accuracy was tested using three different measures: (1) a distance from the center of mass-each ground truth model to the center of mass of each acquired model, (2) an orientation error in the (x,y) plane, and (3) a coverage percentage computed in pixels. Table 2-2 shows the errors for each of the nine regions, and the computed total Residual Mean Square (RMS) error for all the subregions.

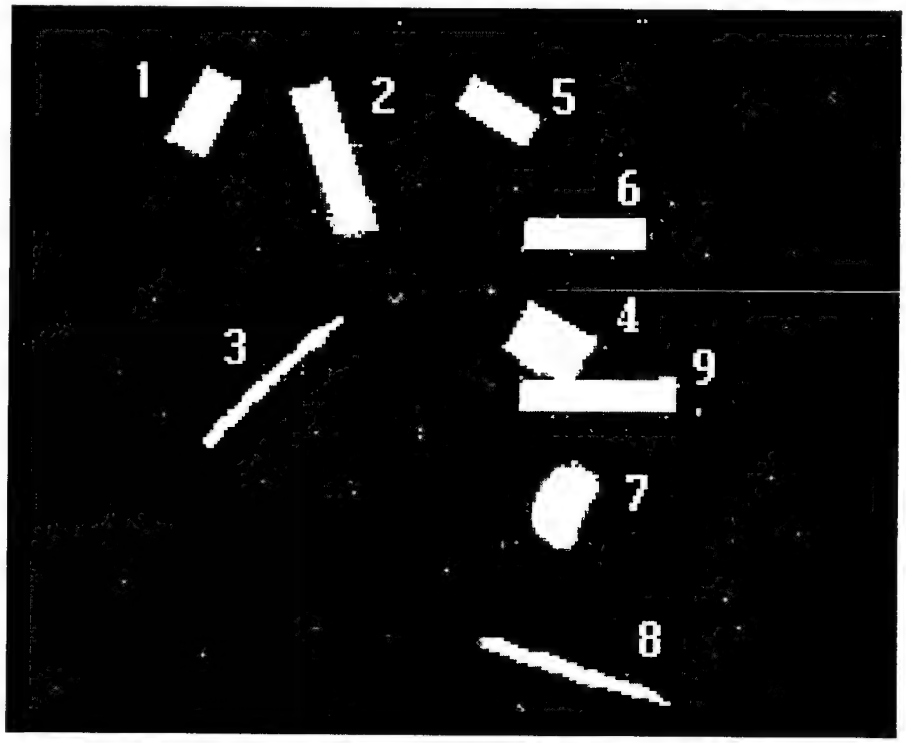
2.5.2 Building Reconstruction Experiment

The second test was performed in the aerial image domain using an elevation map reconstructed from a downlooking stereo pair of the Ascona/ISPRS, "flat" scene. The scene contains five peaked roof buildings of complex rooftop structure. Because building rooftops are expected to be perpendicular to the gravity vector, relative orientations in the model-indexing phase were restricted to rotations about the Z-axis.

The system was run and a local ground plane was fit, producing 12 initial subregions for further processing. Of the 12 subregions, seven remained after processing. Two non-building regions were reconstructed as part of the final scene. A dome with a half base-to-height ratio was reconstructed at the location of a group of trees (see bright circle, top right of Figure 2-9b). A long row of trees was reconstructed as a cylinder with a third base-to-height ratio.



(a)



(b)

Figure 2-7. (a) Range image used for scene reconstruction. (b) Nine detected regions after region 0 (ground plane) has been fit.

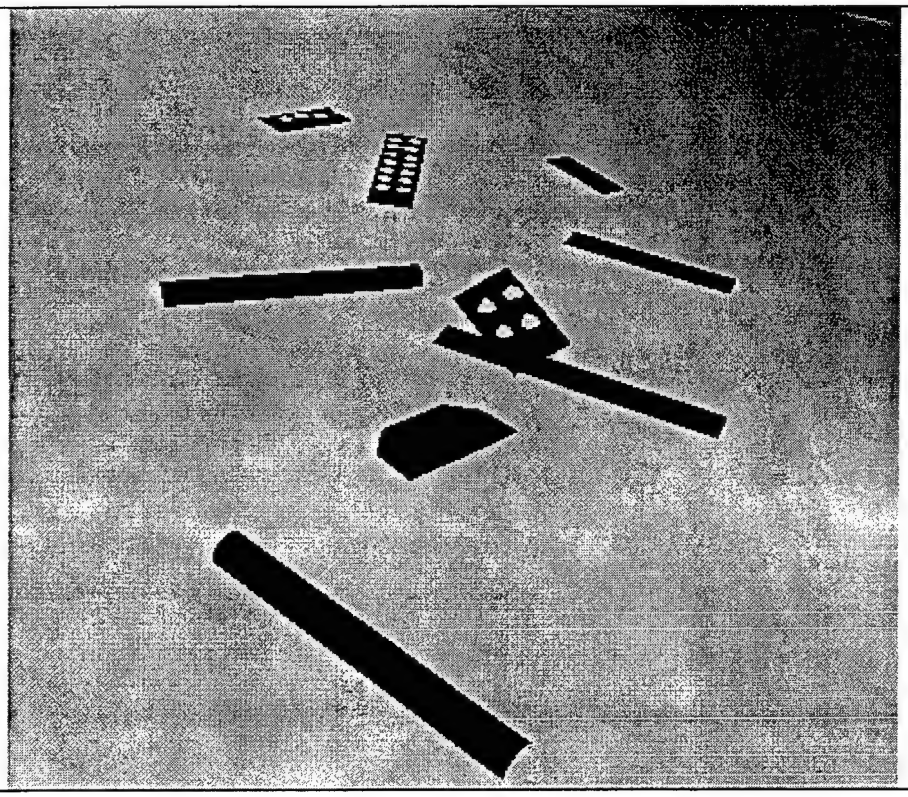
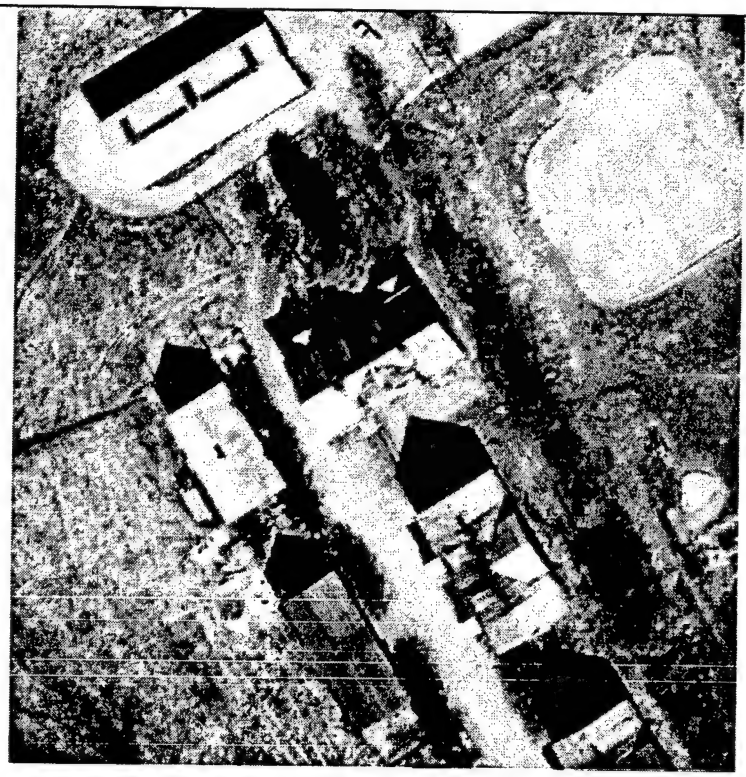


Figure 2-8. Reconstructed surfaces of the "tabletop" scene.

Table 2-2. Results of tabletop reconstruction. Regions 3 and 8 have unusually high overlap error due to the fact that the tips of the pencils were not reconstructed; see text for a description of the errors reported here.

Region	Center of Mass distance error	Orientation error in x-y plane	Coverage percentage on a pixel basis
1	0.169 cm.	0.018	99.8
2	0.081 cm.	0.031	96.5
3	0.322 cm.	0.018	82.1
4	0.299 cm.	0.023	89.6
5	0.172 cm.	0.037	98.5
6	0.065 cm.	0.013	99.5
7	0.449 cm.	0.103	90.3
8	0.417 cm.	0.019	81.2
9	0.171 cm.	0.092	99.4
Sub-region (RMS)	0.209	---	79.2

The final scene is shown in Figure 2-10. Two surface substructures were detected on two different buildings by the recursive model fitting process; both are roof gables. The gable in the foreground of Figure 2-10 more accurately reflects actual scene structure than does the second gable (which is less accurate due to errors in the DEM).



(a)

(b)

Figure 2-9. (a) Image of the Ascona region used for reconstruction. (b) Corresponding DEM recovered from stereo processing.

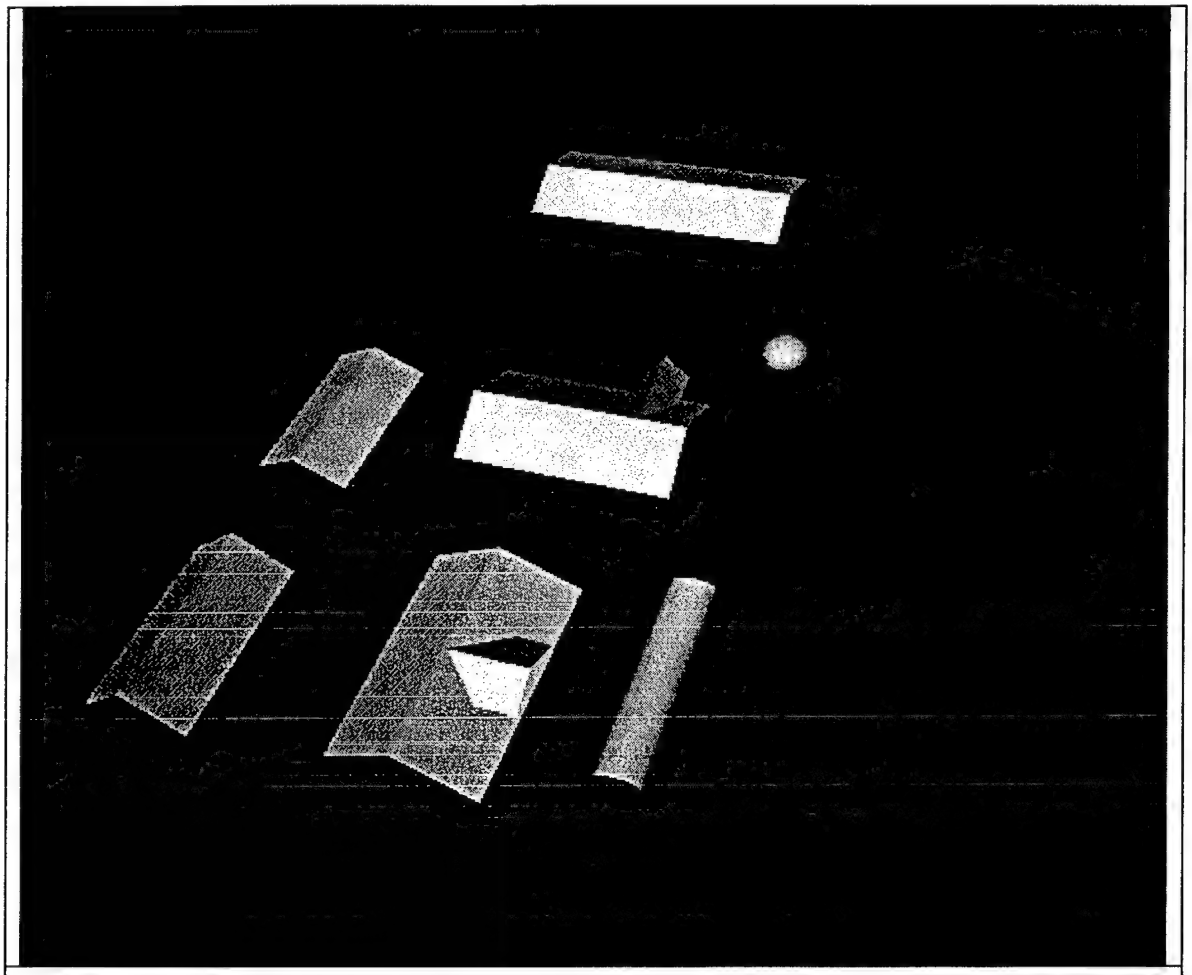


Figure 2-10. Reconstructed scene. All buildings and two rooftop substructures were recovered. Two areas of treetops converged close enough to a cylinder and dome model to be reconstructed.

SECTION 3. RECOVERY OF BUILDING GEOMETRY FROM SAR AND IFSAR

3.1 Introduction

Work at UMass is addressing the issues encountered when attempting to recover geometric structures, such as building boundaries, from high-resolution SAR and IFSAR imagery. The presence of outlier noise and "drop-outs" in such data sets makeS this a unique and challenging task, one markedly different from the interpretation of lower-resolution SAR imagery. The IU approach taken by UMASS to radar interpretation breaks from the traditional, radargrammetric techniques that rely mostly on signal processing (such as square-law constant false alarm rate (CFAR) detection (Weiss 1982)), are that assumptions underlying these methods are often not met by the new, high-resolution data sets (Chellappa et al. 1996a). While others have taken an IU approach as well (see (Chellappa et al. 1996a, Chellappa et al. 1996b)), none have exploited the geometric features recoverable from an IFSAR image to the extent presented here.

Our approach to the problem of extracting rooftop boundaries is twofold. First, one edge of the building is found. Second, we locate the building's back edge, characterized by the shadows that edge casts in the image (Leberl 1990). Next, a portion of the building's rooftop is extracted via region growing. This is accomplished by growing outward from the detected edge, iteratively adding rooftop points adjacent to the growing region until no more can be found. Rooftop points are identified using thresholds found in elevation space. We extract enough of the rooftop so that this information, when combined with the attributes of the back edge detected in the first stage, is enough to fully specify the geometry of the building's boundary. Our initial findings, as presented here, are for buildings with a rectangular boundary, although work is underway for recovering more complex boundary types.

3.2. Profile of the Data Set

SAR is one example of a class of sensors known as side-looking radar (SLR). SLRs gain their name from the fact that they receive and transmit in a direction perpendicular to that of the flight path (i.e., they look to the side of the flight path). A target is imaged only when the ray connecting it to the sensor is normal to the direction of the flight path. The sensor can measure the distance from the target to itself. The strength of the signal returned by the target is measured. Thus, the projection rays of the optical camera are replaced by concentric circles centered at the antenna. IFSAR is used to fully disambiguate the position of an imaged target, generating an orthorectified version of the original SAR intensity image. There are several methods of generating IFSAR data, but we shall only consider the two-antenna, single-pass case here. This means that a single aircraft with two antennas separated by some known baseline collects all the data from the scene in a single pass (the Kirtland AFB and Fort Benning Military Operations in Urban Terrain (MOUT) data sets were collected in this manner).

Both the MOUT and Kirtland data sets have a high frequency of "drop-outs," which are points within the image containing no data. "Drop-outs" are generated by several different phenomena. Some "drop-outs" are due to the fact that no return was measured for that resolution cell. This occurs, for example, when a target on the ground is occluded from the sensor. This also can happen with highly specular targets whose surface normals do not point toward the sensor. "Drop-outs" also can be generated by the orthorectification process, as is the case with so-called "layover holes." "Drop-outs" are both a boon (they make the detection of shadows cast by buildings and other tall objects possible, for example) and a curse (missing data can complicate image interpretation). There also is a significant presence of noise in data sets produced by high-resolution SAR Interferometry. Under these imaging conditions, the intensity returned by a resolution cell on the ground no longer is governed by the complex Gaussian (Rayleigh magnitude) distribution. Instead, a cell's intensity follows a jagged, wildly fluctuating, distribution [2]. This accounts for part of the "noise" observed in both the intensity and phase measurements.

3.3 Back Edge Detection

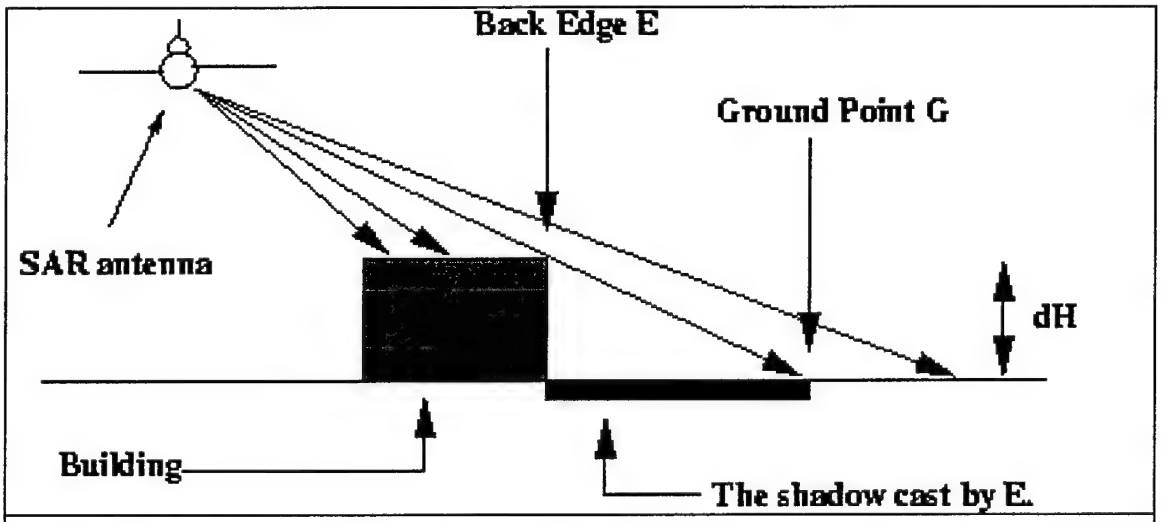


Figure 3-1. Geometry of SAR data acquisition. The shadow cast by a building's back edge extends from back edgel E to a point G belonging to the surrounding terrain

The back edges of a building are along those walls facing away from the sensor. The rooftop of the building occludes the ground adjacent to a back edge from the sensor, causing no return to be measured for that portion of the surrounding terrain. Points belonging to a building's back edge therefore can be identified by the shadows they cast in the image (see Figure 3-1). These shadows extend outward from the back edge in the direction of the sensor. Since the occluded area in this situation is part of the terrain surrounding the building, shadows will terminate at some point on the ground. Thus, a back edgel (which is part of the building's roof) will have an elevation greater than that of the point (which is part of the ground) at the terminating end of its shadow. This information allows the formulation of three different constraints that any point E must satisfy before being labeled a back edgel :

- E must lie on the border between a rooftop and the shadow it casts.

- There must be a shadow extending from E in the direction of the sensor that terminates at some point G, belonging to the surrounding terrain. As such, E must have an elevation greater than that of the surrounding terrain as represented by G (i.e., the height disparity dH between E and G must be large).

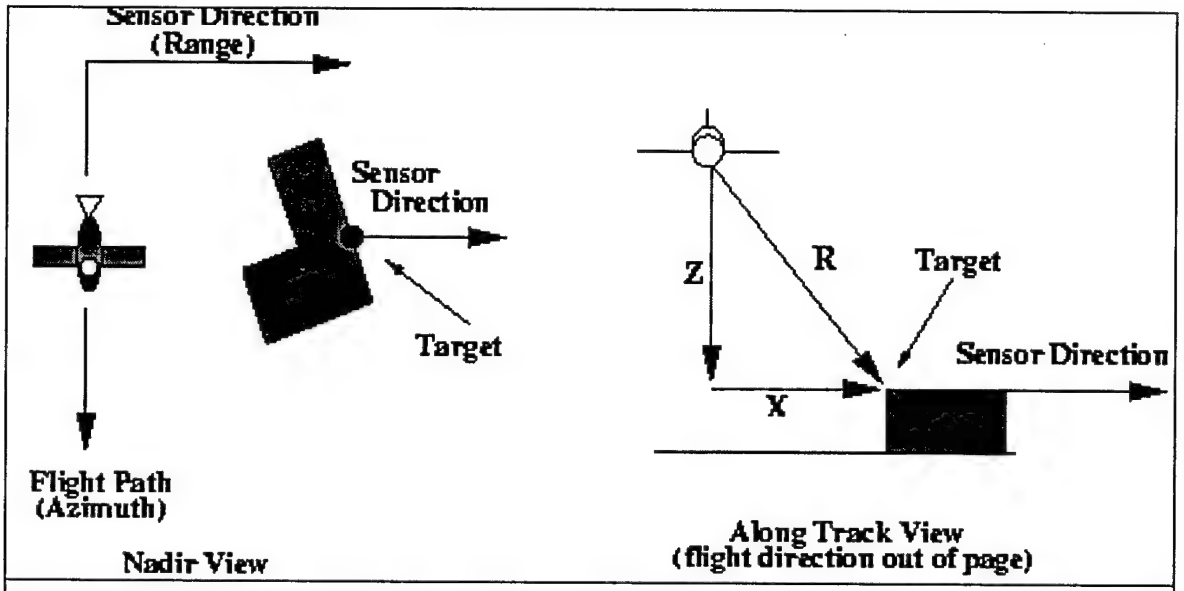
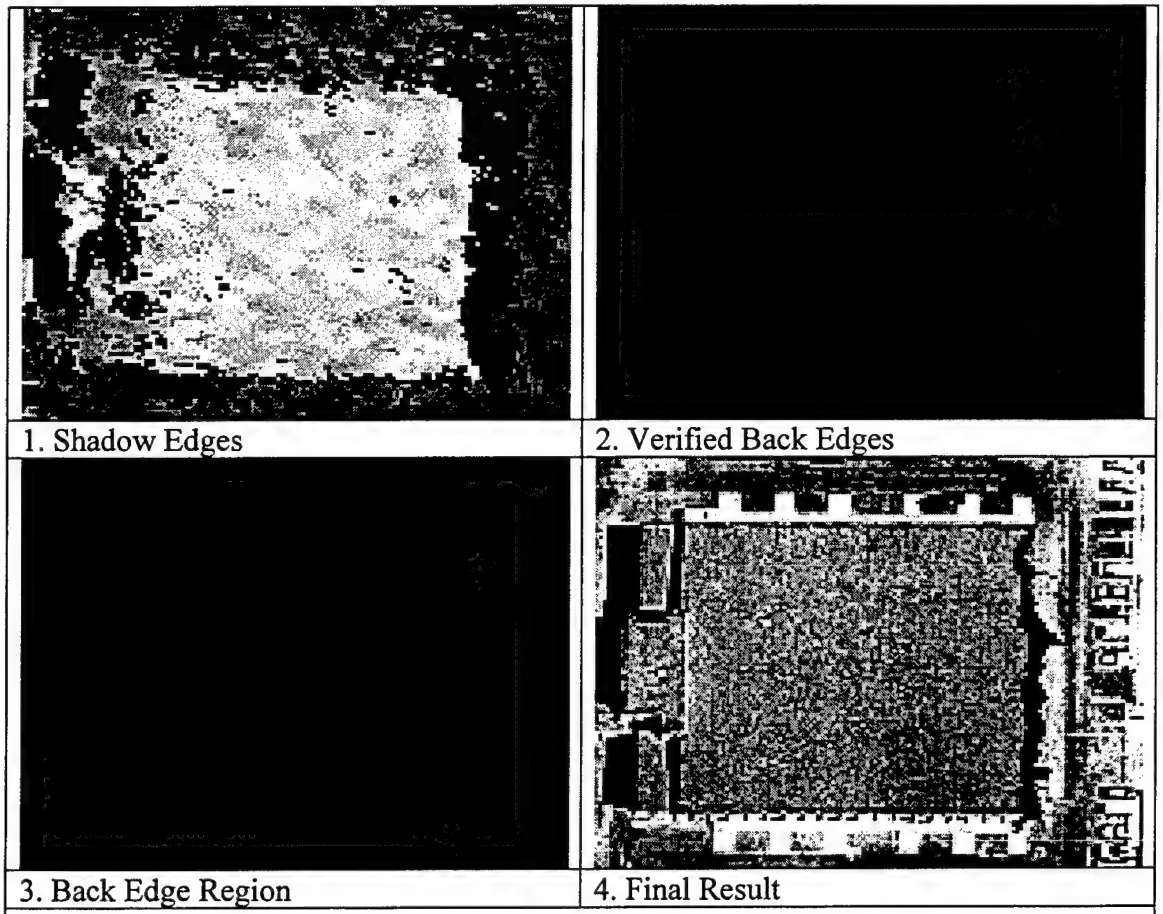


Figure 3-2. Nadir and Along-Track Views of the Data Collection process.

Note that the "direction of the sensor" is the 2-D projection of the ray $\vec{R}$ emanating *from* the sensor and terminating at the target (see Figure 3-2).

The process (see Figure 3-3) begins by identifying those points in the image that satisfy the first constraint. Such points will be referred to as *shadow edges*. Shadow edges in the image are found by applying a series of binary masks to every point in the image for which a return and elevation was measured (see Figure 3-4). The masks are divided into a bright and dark side along a line passing through the mask's center. This border line is shown in red in Figure 3-4. Each mask represents a potential configuration for points that belong to a shadow/rooftop border: the point at the center of the mask is the border element in question; the points on the dark side of the mask lie in the building's shadow; and the points on the bright side of the mask are on the building's rooftop. The border line represents the building edge itself. The masks have border lines at all orientations (in 10 degree intervals), except for those perpendicular to, or face toward, the flight path, as building edges at these orientations would not cast a shadow. This is so that back edges of all orientations can be accommodated. The idea is to then identify rooftop points that border shadowed regions by matching these points and their neighbors to one of the configurations represented by the masks.

The match score for a mask is computed by first determining the error set. The error set are those points within the mask's region of support - here, a 7 by 7 square of pixels centered at the point under test - that disagree with the configuration represented by that mask. A point falling into the region reserved for the shadow cast by the building (i.e., the dark side of the mask) is added to the error set if there is a return for that point, as occluded points cannot register returns with the sensor. A point falling into the region



1. Shadow Edges | 2. Verified Back Edges

3. Back Edge Region | 4. Final Result

Figure 3-3. Stages of the Back Edge detection process.

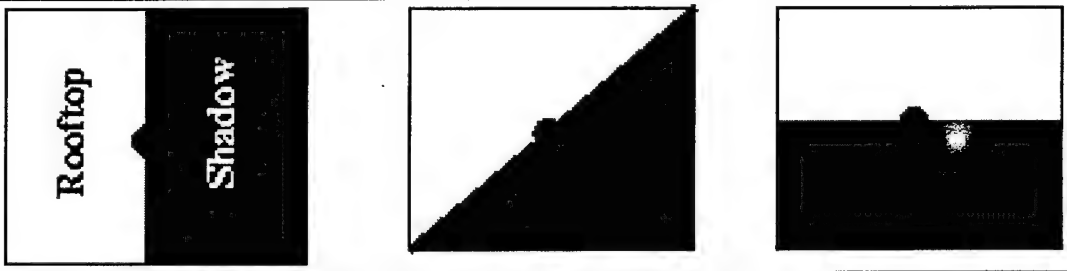


Figure 3-4. Binary masks at varying orientations. These are used to determine if the center point (shown in blue) borders a shadowed region.

reserved for the building's rooftop (i.e., the bright side of the mask) is added to the error set if there is **no** return for that point, since presumably the point is not occluded and it should have returned the emitted signal (note this logic is somewhat flawed since there are other situations in which a target on the ground will not produce a return - see Profile of the Data Set). Once the error set is determined, the distance of each point in the error set from the centerline (shown in red in Figure 3-4) is computed. These distances are summed and divided by the total area of the mask, providing the match score. A point becomes a shadow edge if the application of any of the masks produces a match score below a certain level (the threshold is set once manually for all experiments). The results of the shadow edge detection process is shown in Figure 3-3 (upper left), superimposed on the IFSAR image.

The shadow edges are then verified by searching in the direction of the sensor for the ground point G, presumed to be at the tail end of the shadow cast by the building, then comparing the elevation of that point to that of the candidate's (i.e., enforcing the second constraint). If the candidate's elevation is significantly greater than that of G's, the candidate is confirmed as belonging to a building's back edge. G is located by moving a small window in the direction of the sensor until the majority of the pixels in that window have measured returns. The median elevation value is then selected as G. For each back edgel verified in this way, the ground point G used to verify it is stored along with the edgel E itself for later use in the region growing phase (see Region Growing). The set of points verified as back edgels can be seen in Figure 3-3 (upper right). These points were selected from the shadow edges shown in Figure 3-3 (upper left).

The back edgels so detected are grouped into larger regions that represent entire (or large parts of) building back edges by first interpolating between verified back edgels, then performing a morphological closing to form connected regions of back edgels (Figure 3-3, lower left). A line is fit to the medial axis of these regions using a Hough transform. The orientation of the fitted line (Figure 3-3, lower right) is used as an approximation to the building's overall orientation. Note this property is significant only in the context of rectilinear building boundaries.

3.4. Region Growing

After the building's back edge has been detected, the remainder of that building's boundary is extracted using a region growing technique that identifies points within the same local neighborhood as belonging to either the ground or the building's rooftop, then adds those points identified as rooftop to the growing region. These decisions are made based on a threshold found in an elevation histogram of that local neighborhood (see below). Points falling into the classification window that already have been assigned labels are used to constrain the selection of the threshold. The region's growth is restricted to proceed along those points lying on the rooftop/ground boundary (i.e., those points that have been labeled as rooftop but are adjacent to points labeled as ground). These points are the seeds from which growth proceeds. In this way, the boundary of the building is traced out in the image by the progression of the rooftop region's growth (see Figure 3-5).

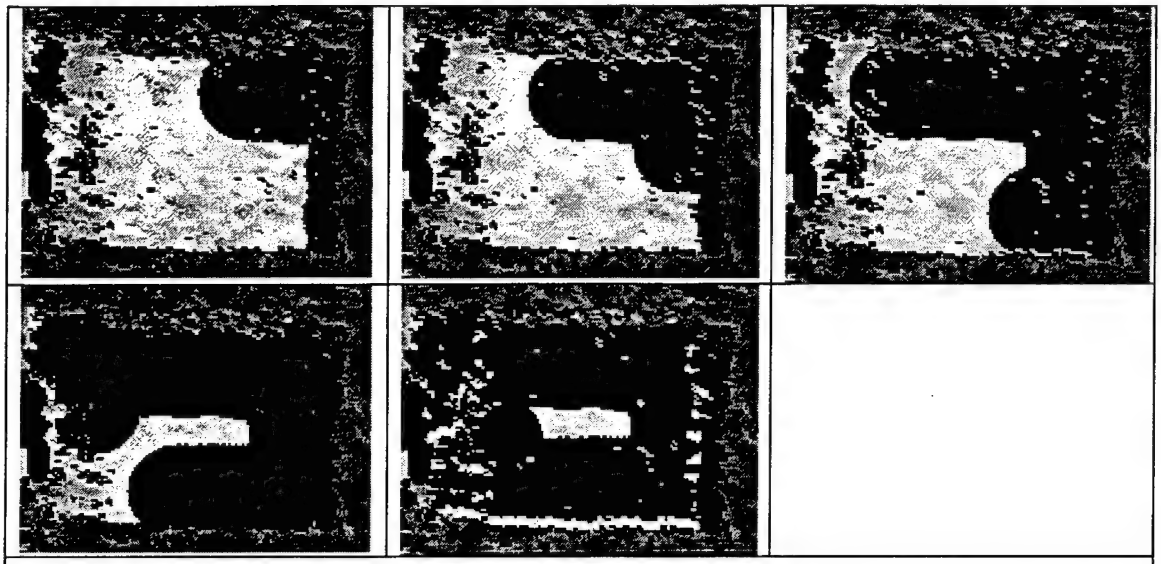


Figure 3-5. Extracting the remainder of the building's boundary via region growing. The rooftop region grown so far is shown in blue, while the back edge (Figure 3-3) from whence it began is in red. The region's growth progresses from left to right, top to bottom.

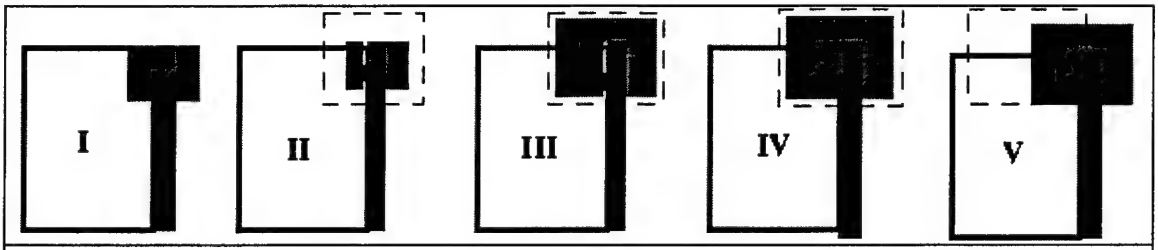


Figure 3-6. A single iteration of the region growing algorithm. Illustrated in red are seed points, in purple the actual building boundary, in green those points labeled as ground and in blue those labeled as roof. The shadow of the back edge is shown in black.

The region growing process is straight-forward (see Figure 3-6). A list of seed points is maintained for the growing region. This list initially contains the points belonging to one of the tips of the back edge. These points are labeled as rooftop, and form the initial region (blue area in Figure 3-6 – left panel). The points from the surrounding terrain used to confirm the initial seeds as belonging to a building back edge (the ground point G referred to in the Detection of Building Back Edges) are labeled as ground (green area in Figure 3-6). The remainder of the image is labeled as unclassified. At each iteration, a new seed point is removed from the list of seeds (shown in red in Figure 3-6 – second panel). Unlabeled points within an adaptively sized window, centered at that point, are to be classified. The window is constrained to be large enough to encompass several points already labeled as ground to aid in the selection of a threshold (dotted box in Figure 3-6 –

second panel). At least 15 such points must fall into the window, but any number of points that allow a reasonably accurate estimation of the mean ground height can be used. An elevation histogram of the unclassified points is taken and a threshold is selected in order to classify the points (Figure 3-6 – third panel). Those newly classified points adjacent to points labeled as ground are added to the list of seeds (shown in red in Figure 3-6 – fourth panel). Region growing proceeds (Figure 3-6 – fifth panel) until the list of seeds is empty, giving us our rooftop region.

Classifications are based on a threshold found in an elevation histogram of the area immediately surrounding a seed point. Since the region's growth is restricted to points near an edge of the building, the local neighborhood of a seed will contain points from both the rooftop and the surrounding terrain. Therefore, an elevation histogram of such a neighborhood should have distinguishable modes corresponding to the rooftop and the ground, allowing a suitable threshold to be found between those modes.

Since the terrain adjacent to a building is typically flat (at least locally), ground points within the same neighborhood should have similar elevations. The elevations of points within the classification window that already have been labeled as ground can be used to predict the elevations of any other ground points within that window that have yet to be classified. Specifically, the mean elevation of the points just recently labeled as ground should not differ from the mean elevation of those points already classified as such, provided both sets of points are near one another. To ensure that both sets of points are near one another, the maximum radius of the classification window is set to a heuristically selected value of 4 m.

After the window size is selected, the mean elevation μ_{prior} of the points already labeled as ground is computed. The local minima of the elevation histogram are then extracted, as these values may represent the nadir found between different modes in a histogram. The set of these elevations S_{minima} serve as potential thresholds. For each elevation E_i in S_{minima}, the mean elevation μ_i of all unclassified points with heights less than or equal to E_i is computed. That is, we compute the mean elevation of the points that would be labeled as ground by that threshold. Once this has been done, our threshold is the elevation E_i in S_{minima} that minimizes the absolute value of $|\mu_{prior} - \mu_i|$. After the threshold has been selected, classification ensues. Note that in the early iterations of the process, the ground points G used to validate the back edgels provide our estimates of the ground's elevation.

An example can be seen in Figure 3-7. There are two local minima in this histogram, indicated in red. The first (minima A) is at 102.672 m, and the second (minima B) is at 104.683 m. The mean elevation μ_{prior} of the points already classified as ground is 102.199 m. The mean ground elevation if minima A was used as the threshold would be 102.002 m, only 0.197 m difference from the prior mean of 102.199 m. The mean ground elevation if minima B was used as the threshold would be 102.96 m, a difference

of 0.761 m from μ_{prior}. Thus, a threshold of 102.762 m is selected, since using A resulted in a smaller difference from μ_{prior} than B.

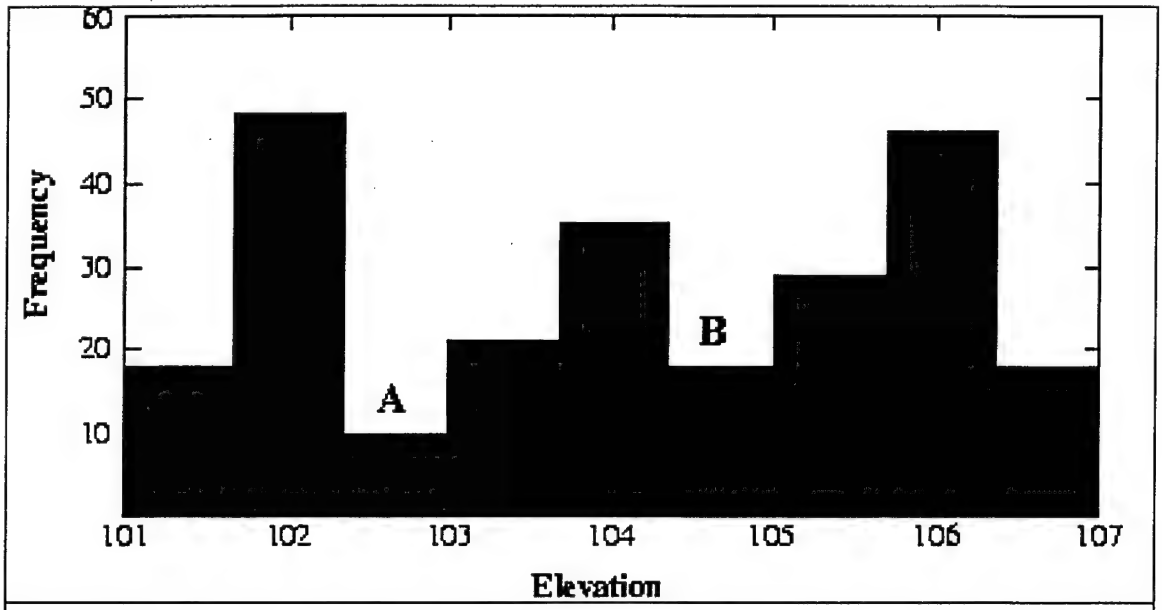


Figure 3-7. Illustrating the selection of an elevation histogram using data derived from the example in Figure 3-5. Minima A is preferred as a threshold over minima B since the difference between ground points classified using A and the mean of the previously classified ground points is smaller with A.

3.5. Final Results

Boundaries were established for each building found in the image by placing a bounding box around the combined back edge/rooftop region found for that building. These bounding rectangles were at an orientation equal to that of the building's back edge. The resultant fits for selected areas of the Kirtland AFB and Fort Benning MOUT sites are shown in Figures 3-8 and 3-9. All three building in the Kirtland AFB scene were detected. Thirteen of the 15 buildings in the MOUT scene were detected. The errors in orientation and coverage have not yet been fully evaluated.

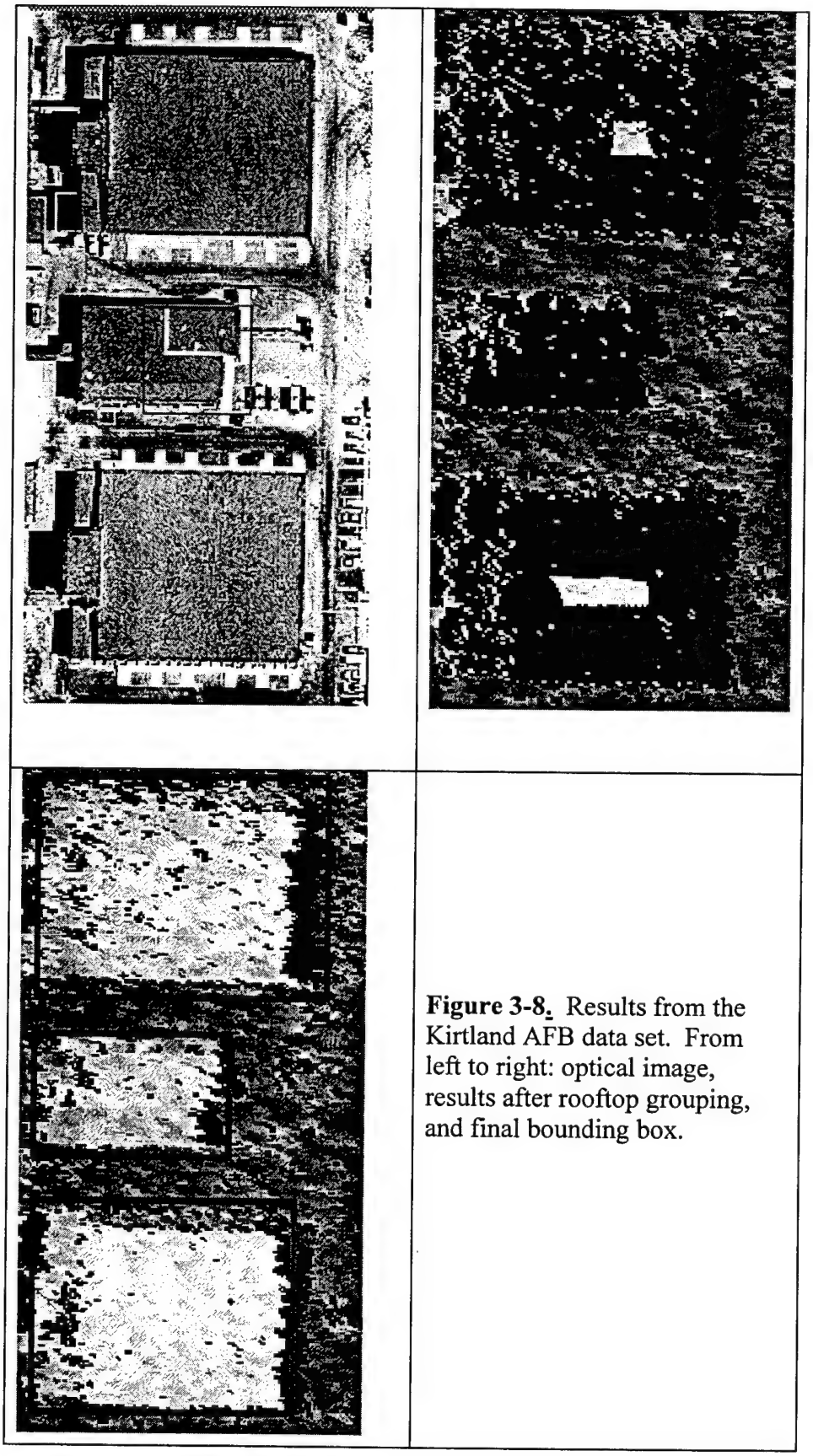


Figure 3-8. Results from the Kirtland AFB data set. From left to right: optical image, results after rooftop grouping, and final bounding box.

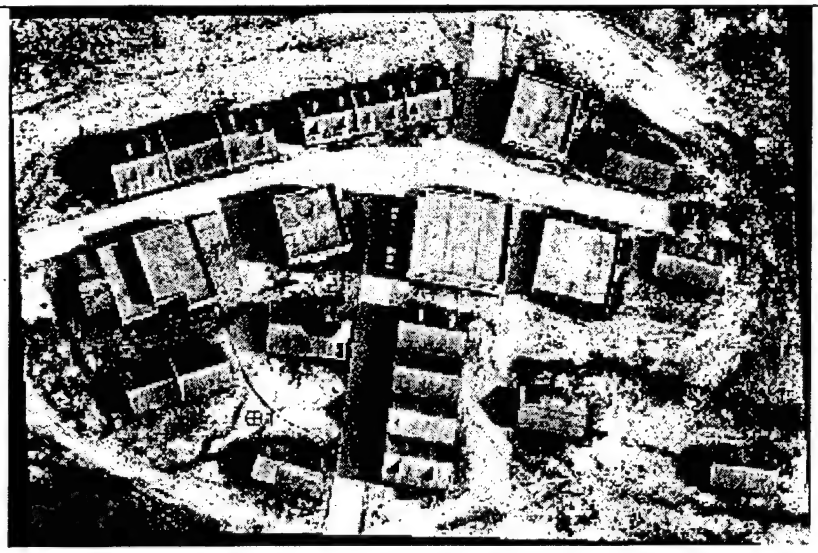


Figure 3-9. Results from the Fort Benning MOUT data set. From top to bottom: optical image, results after rooftop grouping, and final bounding box.

SECTION 4. BIBLIOGRAPHY

Andersen, S., K. Olesen, F. Jensen, and F. J. Hugin, "A Shell for Building Bayesian Belief Universes for Expert Systems," *Proceedings of the 11th International Congress on Uncertain Artificial Intelligence*, pp. 1080—1085, 1989.

Aurenhammer, F., "Voronoi Diagrams - A Survey of a Fundamental Geometric Data Structure," *ACM Computing Surveys*, Vol. 23. pp. 345-405, 1991.

Besl, P. and R Jain, "Segmentation Through Variable Order Surface Fitting," *IEEE T-PAMI*, Vol. 10, pp. 167-192, 1988.

Besl , P. and R. Jain, "Three-Dimensional Object Recognition," *Computing Surveys*, No. 1, pp. 75-145, 1985.

Brooks, R., "Symbolic Reasoning Among 3-D Models and 2-D Images." *Artificial Intelligence* Vol. 17, pp. 285-348, 1981.

Brown, C., M. Marengoni, and ,G. Kardaras, "Bayes Nets For Selective Perception and Data Fusion," *Proceedings of the SPIE on Image and Information Systems: Applications and Opportunities*, 1994.

Chellappa, A. R., S. Kuttikkad, L.M. Novak, "Building 2-D Wide Area Site Models from Single- and Multi-pass Polarization SAR Data," *SPIE Proceedings: Algorithms For Synthetic Aperture Radar Imagery III*, pp. 34 - 44, April 1996a.

Chellappa, A. R., S. Kuttikkad, R. Meth, P. Burlina, K. Ome, C. Shekkar, "Model Supported Exploitation of SAR Imagery.," *Proc. ARPA Image Understanding Workshop*, pp. 389-408, 1996b.

Cohen, I., L. Cohen, and N. Ayache, "Introducing Deformable Surfaces to Segment 3-D Images and Infer Differential Structure," Technical Report, INRIA, 1991.

Collins, R., Y. Cheng, C. Jaynes, F. Stolle, X. Wang, A. Hanson, and E. Riseman, "Site Model Acquisition and Extension from Aerial Images;" *Proceedings of the International Conference on Computer Vision*, pp. 888-893, 1995.

Collins, R., C. Jaynes,Y. Cheng, X. Wang, F. Stolle, A. Hanson, and E. Riseman, "The Ascender System: Automated Site Modeling from Multiple Aerial Iimages," Computer Vision and Image Understanding - Special Issue on Building Detection and Reconstruction from Aerial Images, to appear 1998.

Cooper, G., "The Computational Complexity of Probabilistic Inference Using Bayesian Belief Networks," *Artificial Intelligence*, Vol. 42, pp. 393-405, 1990.

Draper, B., R. Collins, J. Broglio, A. Hanson., and E. Riseman, "The schema system." *International Journal of Computer Vision*, pp. 209-250, 1989.

do Carmo, M. P. *Differential Geometry of Curves and Surfaces*, Prentice-Hall, Englewood Cliffs, N.J, 1976.

Fua, P. "Reconstructing Complex Surfaces from Multiple Stereo Views", *5th International Conference on Computer Vision* , Cambridge, MA, 1995.

Fua, P. and Y. G. Leclerc, "Using 3-Dimensional Meshes To Combine Image-Based and Geometry-Based Constraints," *European Conference on Computer Vision*, pp. 281-291, Stockholm, Sweden, May 1994.

Hanson, A., and E. Riseman, "Visions: A computer system for interpreting scenes." In *Computer Vision Systems*, A. Hanson and E. Riseman, Eds. Academic Press, 1978.

Haralick, R., and L. Shapiro Computer and Robot Vision. Addison-Wesley, 1993.

Herbert,M., K. Ikeuchi, and H. Delingette. "A Spherical Representation for Recognition of Free-Form Surfaces." *IEEE T-PAMI*, Vol. 17, No. 7, pp. 681-689, 1995.

Herman, M., and T. Kanade, T. "Incremental reconstruction of 3-d scenes from multiple complex images." *Artificial Intelligence*, Vol. 30, pp. 289-341, 1986.

Horn, B. *Robot Vision*, MIT Press, McGraw-Hill, Cambridge, MA, 1986.

Jaynes, C., F. Stolle, H. Schultz, R. Collins, A. Hanson, and E. Riseman, "Three-dimensional grouping and information fusion for site modeling from aerial images." *DARPA Image Understanding Workshop*, pp. 479-490, 1996.

Jaynes, C., M. Marengoni, A. Hanson, and E. Riseman, "3-D model acquisition using a bayesian controller." *Proceedings of the International Symposium on Engineering of Intelligent Systems*, Tenerife, Spain, to appear 1998.

Jaynes, C., A. Hanson, E. Riseman, and H. Schultz. "Building Reconstruction from Optical and Range Images," *International Conference on Computer Vision and Pattern Recognition (CVPR'97),* San Juan, Puerto Rico, 1997.

Jaynes, C., F. Stolle, and R. Collins, "Task driven perceptual organization for extraction of rooftop polygons," IEEE Workshop on Applications of Computer Vision, pp. 152-159, 1994.

Jensen, F. An introduction to Bayesian networks. Springer Verlag, NY,NY, 1996.

Leonardis, A., A. Gupta, and R. Bajcsy. "Segmentation of Range Images as the search for geometric parametric models," *Int. Journal of Computer Vision*, Vol. 14, pp. 253-277, 1995.

Lowe, D., "Fitting Parameterized three-dimensional models to images," *IEEE T-PAMI*, Vol. 13, pp. 441-450, 1991.

Kass,M., A.Witkin, D. Terzopoulos. "Snakes: Active Shape Models," *Int. Journal of Computer Vision*, Vol. 1, pp. 321-331, 1988.

Kumar, V., and U. Desai, "Image interpretation using bayesian networks." *IEEE T-PAMI*, Vol. 18, No. 1, pp. 74-77, 1996.

Leberl, A. F. W., Radargrammetric Image Processing, Artech House, pp. 77 - 89, 1990.

McKeown, D., Jr., A. W., and J. McDermott, "Rule-based interpretation of aerial imagery." *IEEE T-PAMI*, Vol. 7, pp. 570-585, 1985.

Miller, J., and C. Stewart. "Prediction Intervals for Surface Growing Range Segmentation," *International Conference on Computer Vision and Pattern Recognition (CVPR'97)*, San Juan, Puerto Rico, 1997.

Montagnat, J., and H. Delingette. "A Hybrid Framework for Surface Registration and Deformable Models," *International Conference on Computer Vision and Pattern Recognition (CVPR'97),* San Juan, Puerto Rico, 1997.

Nelder, J., and R. Mead, *Computational Journal*, vol. 7, pp. 308-313, 1965.

Pearl, J. *Probabilistic Reasoning in Intelligent System: Networks of Plausible Inference.* Morgan Kaufmann, 1988.

Rimey, R., and C. Brown, "Task-oriented vision with multiple bayes nets." In Active Vision, B. A. and Y. Aliomonos, Eds. The MIT Press, 1992.

Strat, T. "Employing contextual information in computer vision." *Proceedings of ARPA Image Understanding Workshop*, 1993.

Taubin, G. . "Estimation of planar curves, surfaces, and non-planar space curves defined by implicit polynomials with applications to edge and range image segmentation," *IEEE T-PAMI*, Vol. 13, No. 11, pp. 1115-1138, Nov. 1991.

Terzopoulos, D., and D. Metaxas, "Dynamic 3-D Models with local and global deformations: Deformable Superquadrics," *Int. Conference on Computer Vision (ICCV'90),* pp. 600-615, Osaka, 1990.

Wang, C., and S. Srihari, "A framework for object recognition in a visually complex environment and its application to locating address blocks on mail pieces." *International Journal of Computer Vision*, pp. 125-151, 1988.

Weiss, A. M., "Analysis of Some Modified CFAR Processors." *IEEE Transactions on Aerospace and Electronic Systems*, pp. 102 - 114, Jan. 1982.

Zhang, Z., "Iterative point matching for registration of free-form curves and surfaces." *Int. Journal of Computer Vision*, Vol. 13, No. 2, pp. 119-152, Dec. 1994.

Zhang, D., and M. Herbert, "Multi-Scale Classification of 3-D Objects," *International Conference on Computer Vision and Pattern Recognition (CVPR'97),* San Juan, Puerto Rico, 1997.

CVPR '97

Building Reconstruction from Optical and Range Images*

Christopher O. Jaynes, Allen Hanson, Edward Riseman, and Howard Schultz
Computer Vision Laboratory
Dept. of Computer Science,
Box 34610 University of Massachusetts, Amherst

Abstract

A technique is introduced for extracting and reconstructing a wide class of building types from a registered range image and optical image. An attentional focus stage, followed by model indexing, allows top-down robust surface fittting to reconstruct the 3D nature of the buildings in the data. Because of the effectiveness of model selection, top-down processing of noisy range data still succeeds and the algorithm is capable of detecting and reconstructing several different building roof classes, including flat single level, flat multi-leveled, peaked, and curved rooftops.

The algorithm is applicable to range data that may have been collected from several different range sensor types. We demonstrate reconstructions of different buildings classes in the presence of large amounts of noise. Our results underline the usefuless of range data when processed in the context of a focus-of-attention area derived from the monocular optical image.

1 Introduction

We introduce a solution to the problem of building reconstruction from aerial images. The technique presented here supports the reconstruction of a wide class of building models and is robust to sensor noise. The reconstructed models may be used for urban planning, three dimensional visualization for simulated walk/fly throughs, and as a geometric representation capable of supporting additional image understanding tasks.

The detection and accurate reconstruction of three dimensional scene structures from aerial data proves to be difficult in both optical and range data. Within optical imagery, building boundaries are occluded by other features at the site and rooftop regions contain varying surface structure and surface markings. Problems in range images are similarly complex. Depth discontinuities at the boundaries of buildings cause typical correlation-based stereo optical systems to degrade. Likewise, depth computed from radar depends on surface geometry and material properties of objects in the scene.

A successful automated reconstruction system will make use of both range and optical data in order to overcome the inherent problems in each. The system should be robust with respect to a wide variety of range sensors including Interferometric Synthetic Aperature Radar (IFSAR) and optical stereo. This paper introduces one such system that demonstrates how appropriate use of both the registered optical image and range data increases reconstruction accuracy, reliability and completeness.

There has been a large amount of work in aerial image interpretation, and building reconstruction in particular, using both monocular and multiple image strategies [Collins, Jaynes, et. al '96, Jaynes'94, Herman'94, Matsuyama'85, McKeown'90]. In earlier work, perceptual organization techniques provided the impetus for many building extraction systems. Heurtas and Nevatia [Huertas'80], organize lines and corners into possible rectangles and select the best possible set of groupings from the hypotheses. The ASCENDER system [Collins'95] hypothesized 2D building rooftops through a perceptual grouping scheme [Jaynes'94] and computed a height estimate for each roof through multi-image triangulation. The system assumed flat roofed buildings and extruded the rooftop polygon to a known ground plane.

More recently, the utility of range images for site reconstruction has been recognized. Kim and Muller [Kim'95] extract rooftop boundaries from optical data and use an elevation map to estimate rooftop height. Haala and Hahn [Haala'95] search an elevation map for local maxima, with three dimensional lines computed in these regions used for parametric model fits to the extracted line segments. The approach estimates the initial parameters for model fitting, but assumes that the buildings at the site can be reconstructed using a single parametric model (e.g. peaked roof model).

All these approaches assume a small class of building types (typically flat roofs), and, in the case of the ASCENDER system, where no elevation map is utilized, require several registered optical images to ar-

*Funded by the RADIUS project under DARPA/Army TEC contract number DACA76-92-C-0041, by NSF grant number CDA-8922572.

rive at an accurate 3D model. The work presented here makes few assumptions about the shape of the building rooftop, other than it can be composed from a set of surface shapes defined in an existing database.

Our approach to the problem has been threefold: 1) We focused on extraction techniques that were not restricted to a small class of buildings. Instead, automatic classification of the different surface primitives are combined to reconstruct a wide class of building types. Examples are multi-level flat roofs (or single level structures with significant substructures such as air conditioner units), peaked roof buildings, curved-roof buildings, such as Quonset huts or hangars. 2) Our goal is to acquire 3D models even at the limits of feasibility due to noise and the size of the structures being extracted. In order to accomplish this, the use of top-down model application is used to arrive at a correct reconstruction in the presence of significant noise and ambiguities. We make use of a database of surface types that represent geometrically feasible buildings (51 models, in eight classes). 3) The system is fully automatic. Although user intervention can be a valuable source of information, the system attempts to segment, classify, and reconstruct the site completely automatically with as little sensitivity to parameter settings as possible.

2 Segmentation

Segmentation takes place in the optical image using a perceptual grouping scheme first presented in the ASCENDER I system [Collins'95]. The segmented regions from the optical image provide a focus of attention for surface reconstruction within the range image. Figure 1 shows the 450x450 pixel optical and registered range images for an area located at Fort Hood, Texas that will be used to demonstrate the reconstruction process. The range image was constructed using the UMass TERREST [Schultz'94] system from an optical pair of the region.

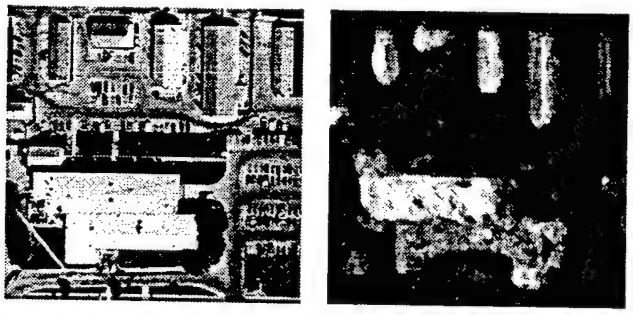


Figure 1: An optical image and corresponding stereo range data for an area of Ft. Hood Texas. Elevation values at right are coded as image brightness.

Segmentation of rooftop boundaries is based on a perceptual grouping scheme [Jaynes'94] that has been shown to be effective in delineating rooftop boundaries in aerial images [Collins, Jaynes, et. al '96]. It is important to note, however, that alternative segmentation schemes can be used to focus the reconstruction process.

Low level features in the segmentation module are straight line segments and corners. We assume that significant rooftop surfaces can be delineated with a flat rectilinear polygons. This implies a search for polygons made up of straight line segments and orthogonal corners (although orthogonal corners in the world are not necessarily orthogonal in the scene when oblique views are processed). To determine a set of relevant corner hypotheses, pairs of line segments with spatially proximate endpoints are grouped together into candidate image corner features. Using the known camera pose, each potential image corner is then back-projected into a nominal Z-plane in the scene, and the hypothetical *scene corner* is tested for orthogonality.

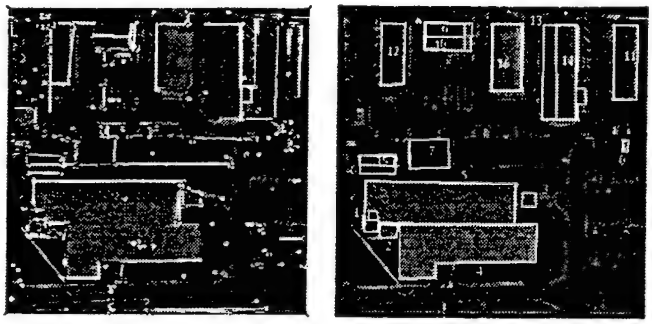
Geometrically, collated features are sequences of geometrically grouped corners and lines that form a chain (Figure 2). Chains are a generalization of the collated features in earlier work [Huertas'80] and allow final polygons of arbitrary rectilinear shape to be constructed from the low level features.

Collated feature chains are represented by paths in a feature relation graph. Low level features (corners and line segments) are nodes in the graph, and perceptual grouping relations between these features are represented by edges in the graph. Nodes have a certainty measure that represents the confidence of the low level feature extraction routines; edges are weighted with the certainty of the grouping that the edge represents. A chain of collated features inherits an accumulated certainty measure from all the nodes and edges along its path.

High Level Polygon hypothesis extraction proceeds in two steps. First, all possible polygons are computed from the collated features. Then, polygon hypotheses are arbitrated in order to arrive at a final set of non-conflicting, high confidence rooftop polygons (Figure 2). All of the cycles in the feature relation graph are searched for in a depth first manner, and stored in a dependency graph where nodes represent complete cycles (rooftop hypotheses). Nodes in the dependency graph contain the certainty of the cycle that the node represents. An edge between two nodes in the dependency graph is created when cycles have low level features in common. The final set of non-overlapping rooftop polygons is the set of nodes in the dependency graph that are both independent (have no edges in common) and are of maximum certainty.

3 Classification

The segmentation module produces a set of two-dimensional closed regions within the optical data that represent high-confidence building rooftop hypothe-



(a) (b)

Figure 2: (a) Perceptually grouped corner and line segments that are represented in a feature relation graph. (b) The final maximally weighted set of closed cycles.

ses. To reconstruct the 3D structure, parameterized models (called surface primitives) are fit to each corresponding region in the registered range data independently. The final buildings are reconstructed as a composition of these primitives. The work here presents the case when each building is modeled as a separate primitive; however, work is underway to merge multiple models into a single complex building. The key to a reconstruction system that is able to address a wide class of building types lies in the system's ability to automatically classify the type of surface primitive associated with each region, and to constrain the parameters of a robust fit (see Section 4) sufficiently to arrive at an accurate solution. Furthermore, the method must be invariant with respect to translation, scale, and noise.

The classification scheme indexes into a database of surface primitives based on an analysis of differential geometry within each region in the range image. The surface orientations of small surface patches are estimated and an orientation histogram is constructed that is then correlated with an existing library of roof models. These orientation histograms, sometimes called the Extended Gaussian Image, are normalized so that they are both scale and translation invariant. A detailed introduction to the Extended Gaussian Image can be found in [Horn'86].

The surface primitive database (SPD) contains a set of surface classes called *surface primitives*, such as planes, cylindrical surfaces, peaks, and spires, known to typically be part of rooftop surfaces. Associated with each surface primitive are a number of models, representing different parameterizations of each class of surface primitives. For example, the "Peak" surface primitive class is the canonical shape for a number of models in the SPD, each with a different peak angle. Corresponding orientation histograms are stored in the SPD for indexing purposes. Figure 3 shows the SPD used for the results shown here. It contains 8 different surface primitives and 51 total models.

For each of the segmented regions, an orientation histogram is constructed and correlated with the set of models stored in the SPD. The set of points within a region are triangulated into a simple surface using the Delauney algorithm [Aurenhammer'91]. The triangulated surface is a set of triangular surface patches, $T_I = (p_1, p_2, p_3)$, where p_1, p_2, p_3 are datapoints from the original pointset. Figure 4 shows the triangulated surface that corresponds to polygon 11 in Figure 2b.

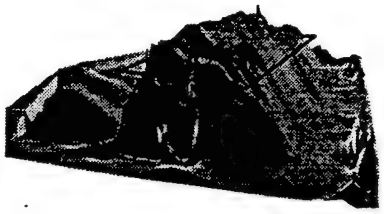


Figure 4: The Delauney representation of the elevation data. The polygon shown corresponds to the upper-rightmost polygon (polygon 11) from figure 2.

The surface normal at each triangular patch is then computed as the cross product of the Vectors $v_1 = \frac{(p_2-p_1)}{\|p_2-p_1\|}$ and $v_2 = \frac{p_3-p_1}{\|p_3-p_1\|}$. Because we assume the normal of the plane representing the footprint of the rooftop is aligned with the gravity vector, the surface normal pointing in the positive Z direction is used to determine the cell on the Gaussian sphere that will receive a "vote" for a particular orientation.

To avoid sensitivity problems with the method in which orientation space is discretized, votes are smoothed over the sphere via a Gaussian function. If the surface normal, N, intersects the Gaussian sphere at (x, y, z), the weighted vote at B is given by:

$$V(x, y, z, B) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(D^2/\sigma^2)} \qquad (1)$$

where D is the angular distance from (x, y, z) to the center of the histogram bucket, B, to receive the weighted vote. The amount of smoothing is related to the expected noise in the range image, however as σ increases the separability of the model classes degrades. For the results shown here, $\sigma = 0.3$ and the orientation histogram contains 240 buckets, reflecting a tesselation based on the semiregular icosahedron.

A single surface normal may induce a smoothed vote over several buckets, given by equation 1, and voting for a given vector stops when the bucket value of $V(x, y, z, B)$ falls below a threshold (0.1 for the results shown here).

Figure 5 shows the histogram constructed from the range data corresponding to polygon 11 in the Fort Hood example. Histograms visualized in this way provide an interesting characterization of the noise within

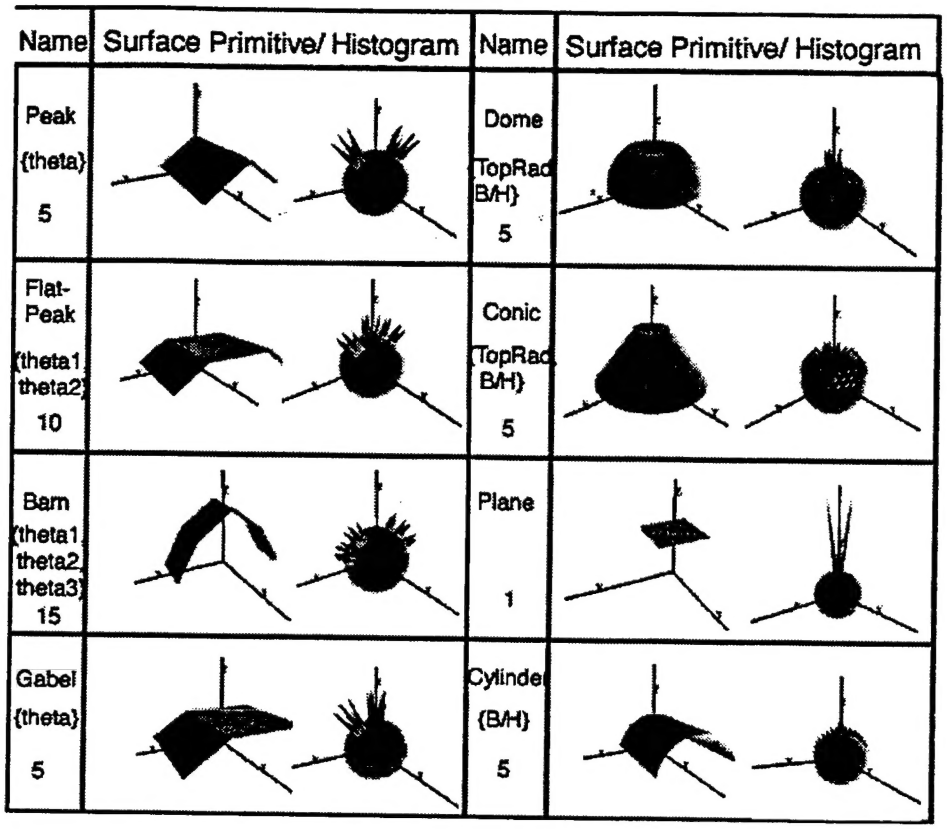


Figure 3: The Surface Primitive Database used for indexing. Each cannonical surface primitive is shown along with the orientation histogram (see text). The parameterization for each model is shown to the left. Several different models, each with unique parameters, are stored in the database. The number of different models is shown to the left.

a surface patch. Notice, for example, that the underlying peak structure is discernable, but that number of votes for each of the "lobes" are unequal.

Figure 5: The surface orientation histogram for the surface contained within the polygon shown in figure 4.

To achieve model indexing, the constructed histogram, referred to as the image histogram, is then correlated with each of the model histograms stored in the SPD. The correlation process is thus independent of scale. The normalized cross-correlation score, given by:

$$C_\theta(I,M) = \frac{\sum^{(i,j)}(I(i,j)-\mu_I)(M(i,j)-\mu_M)}{(\sigma_I * \sigma_M)} \quad (2)$$

where μ and σ represent the mean and the variance respectively.

The method not only selects the correct surface model class (peak roof, for example) based on the correlation score, but estimates the set of parameters for surface fitting (angle of the peak roof). To select the correct model orientation on the ground plane, the value of $C_\theta(I,M)$ must be computed for many possible values of θ that represent different alignments between the spherical histograms I and M. For the results in this paper, θ was restricted to rotations about the Z-axis under the gravitational alignment assumption. The plane model, however, was allowed any possible orientation to reflect the possible presence of flat, sloped roofs or a sloping ground plane. The value of θ for which C is a maximum is stored and compared to the results of other correlations in the database. The highest scoring models are used for robust fitting during the final reconstruction phase. For the results here,

Model Name	Correlation Score	
Peak (130)	0.8813	$\theta = 0.0$
Peak (150)	0.8320	$\theta = 0.0$
Flat Peak (65,65)	0.8054	$\theta = 0.0$
Flat Peak (75,75)	0.7900	$\theta = 0.0$

Table 1: Polygon 11 Correlation Match Scores

the top two models are selected and fit to the data; a residual fit error is used to select the final model.

Table 1 below shows the results of correlating the histogram shown in figure 5 (constructed from polygon 11) with the SPD. The correlation score and the rotation angle of maximum response are shown for the top four models. Note that all are at $\theta = 0.0$, i.e. aligned with the Y axis. A graph depicting the correlation score of the model "Peak (130)" with the histogram produced from the polygon 11 range data is shown in figure 6. Figure 7 shows the model selected for each of the regions in the Fort Hood image.

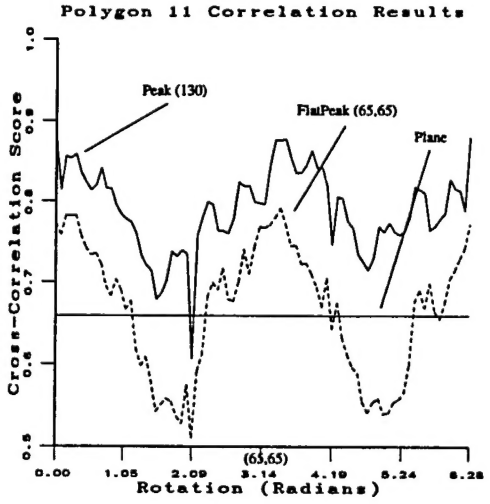


Figure 6: The correlation response of three models (Peak, FlatPeak, and Plane, for comparision) with polygon 11. The highest scoring model other than a peaked roof primitive was the "FlatPeak (65,65)" model. The correlation score, however, clearly separates the two.

4 Reconstruction

Each region within the data has been indexed into the SPD to provide a set of inital models and parameters; these are then fit to the elevation data. The role of the reconstruction module is to use the initial parameters from the SPD match to determine a precise model fit to the range data.

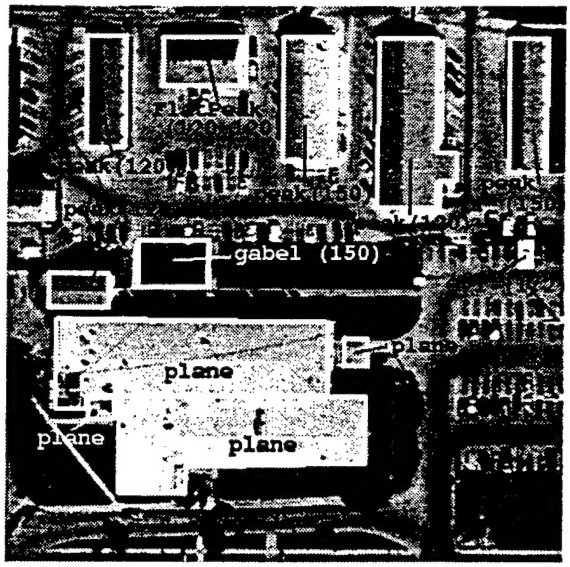


Figure 7: Classification of each region. The model with a maximum response from the library is shown.

For each polygon boundary, the set of elevation points that project within the polygon are extracted from the corresponding range image. All these points are considered to be part of the inital inliers. The model selected from the SPD is fit to the data using a least-median fit technique and the downhill Simplex method.

The downhill Simplex method requires a set of inital parameters to initalize the search; these are provided directly from the indexed SPD model. In order to avoid unusually high residual errors in the case of models with steep surfaces, residual errors are computed as the distance along the approximate surface normal from the elevation data (derived from the Delauney triangulation) to the current model estimate.

Figure 8 shows the final reconstruction after the SPD model "Peak (130)" was fit to the data. The final peak angle (measured from plane to plane) converged to 134 degrees with a median residual error of 0.065 meters for a roof whose height was by 5.9 meters above the groundplane.

Each of the regions was fit to the appropriate SPD model from the database. All remaining points in the range image were assumed to lie in a ground plane and a plane was fit to determine the correct model. Figure 9 shows a rendered view of the entire site model.

Using the registered optical image, a texture map can be wrapped onto each of the rooftop surfaces for better visualization of the site. This was applied to the two flat roof models in Figure 9.

Figure 8: Closeup view of reconstructed polygon 11 (building in foreground).

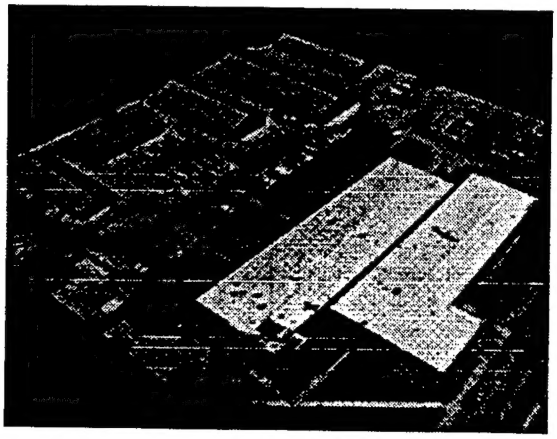


Figure 9: Rendered view of the Fort Hood Scene.

5 Test Results

The technique has also been applied to the Ascona/ISPRS "Flat Scene"; this scene contains several peaked roof buildings of different slopes (see Figure 10). In the face of space restrictions, only key steps in the reconstruction process are shown.

The buildings detected are shown in figure 11. Overlapping polygons were eliminated leaving sixty four percent of the building rooftops to be passed to the indexing module for classification. Note that polygon 21 was detected but was eliminated from the reconstruction process because corresponding range data was not available. The remaining twenty regions were classified using the SPD shown in section 2. The results of the classification are shown in table 2.

The top two models selected for each region were fit to the data and the best fit was chosen for the final reconstruction. All points outside the twenty polygons are used to fit a local ground plane. Planar regions that were classified as planes and were close to the local groundplane (for example polygon 3) were eliminated as false positives and adjusted to lay within the groundplane. Two errors not eliminated by this process are represented by polygons 5 and 7, both being

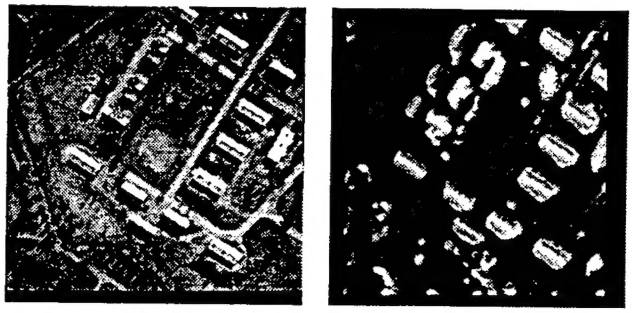


Figure 10: Optical and range image for the Ascona/ISPRS scene.

sloping portions of the rooftops, and reconstructed as extruded planes. Figure 12 shows the final site model acquired as a result of the reconstruction process.

References

[Aurenhammer'91] F. Aurenhammer, "Voronoi diagrams - a survey of a fundamental geometric data structure". *ACM Computing Surveys* 1991. vol 23. pp. 345-405.

[Collins'95] R.Collins, Y.Cheng, C.Jaynes, F.Stolle, X.Wang, A.Hanson and E.Riseman, "Site Model Acquisition and Extension from Aerial Images," *International Conference on Computer Vision*, Cambridge, MA, June 1995, pp. 888–893.

[Collins, Jaynes, et. al '96] R. Collins, C. Jaynes, Y. Cheng, X. Wang, F. Stolle, H. Schultz, A. Hanson, E. Riseman. "The UMass Ascender System for 3D Site Model Construction", chapter to appear in forthcoming DARPA IU RADIUS book. Oscar Firschien (ed), 1996.

[Haala'95] N. Haala and M. Hahn, "Data fusion for the detection and reconstruction of buildings," *International Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Ascona, Switzerland, April 1995.

[Herman'94] M. Herman and T. Kanade. "3D Mosaic Scene Understanding System: Incremental Reconstruction of 3D Scenes from Complex Images". *Proc. ARPA Image Understanding Workshop*, 1994.

[Horn'86] B. Horn. *Robot Vision*, MIT Press, McGraw-Hill, Cambridge, Massachusetts, 1986.

[Huertas'80] A. Huertas and R. Nevatia. "Detecting Buildings in Aerial Images" *Computer Vision, Graphics, Image Processing.* vol. 13, 1980.

[Jaynes'94] C.Jaynes, F.Stolle and R.Collins, "Task Driven Perceptual Organization for Extraction of

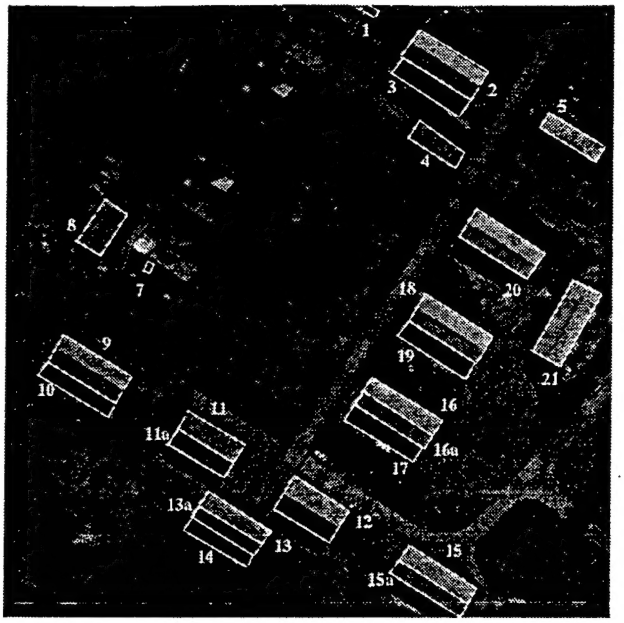


Figure 11: Detected polygons in the Ascona scene. Polygons labeled with a subscript were eliminated due to overlap.

Polygon #	Model Name	Score	θ
1	Plane	0.9322	
2	Peak (120)	0.9104	$\theta = 121.0$
3	Plane	0.9051	
4	Plane	0.9462	
5	Plane	0.8933	
7	Plane	0.7543	
8	Peak (110)	0.8963	$\theta = 29.0$
9	Peak (140)	0.9502	$\theta = 118.0$
10	Plane	0.9353	
11	Peak (140)	0.9017	$\theta = 119.0$
12	Peak (120)	0.8991	$\theta = 120.0$
13	Peak (150)	0.9136	$\theta = 118.0$
14	Plane	0.8773	
15	Peak (120)	0.9114	$\theta = 119.0$
16	Peak (130)	0.9276	$\theta = 116.0$
17	Plane	0.8994	
18	Peak (130)	0.9212	$\theta = 117.0$
19	Plane	0.9102	
20	Peak (120)	0.9532	$\theta = 118.0$

Table 2: Model indexing results for each region in the "Flat" scene.

Rooftop Polygons," *IEEE Workshop on Applications of Computer Vision,* Sarasota, FL, December 1994, pp. 152–159.

[Kim'95] "Building Extraction and Verification from Spaceborne and Aerial Imagery using Image Understanding Fusion Techniques," *International Workshop on Automatic Extraction of Man-Made Objects from Aerial and Space Images*, Ascona, Switzerland, April 1995.

[Matsuyama'85] T. Matsuyama and V. Hwang, "SIGMA: A Framework for Image Understanding: Integration of Bottom-Up and Top-Do wn Processes," *Proceedings of the Ninth IJCAI*, Los Angeles, CA, pp. 908-915, 1985.

[McKeown'90] D. McKeown, "Toward Automatic Cartographic Feature Extraction," *Mapping and Spatial Modelling for Navigation,* Nato ASI Series, Vol. F65, p p. 149–180, 1990.

[Schultz'94] H. Schultz, "Terrain Reconstruction from Oblique Views," *Arpa Image Understanding Workshop,* Monterey, CA, Nov 1994, pp. 1001–1008.

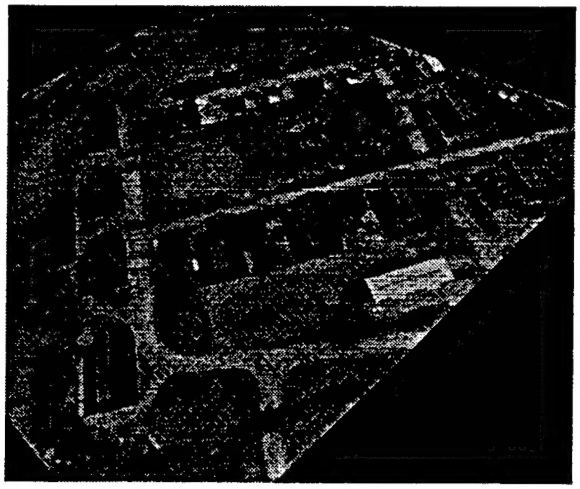


Figure 12: The reconstructed site elevation image.